

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE
UNIVERSITE AKLI MOAND OULHADJE-BOUIRA



Faculté des Sciences et des Sciences Appliquées
Département Informatique

Mémoire de fin d'étude

Présenté par :

SETTI Rabah

BELKACEM Amir

En vue de l'obtention du diplôme de **Master 02** en :

Filière : Informatique

Option : ISIL

Thème :

**Consommation intelligente de l'énergie dans les réseaux de capteurs sans fil
(WSNs)**

Devant le jury composé de :

Mme Aid	Maître de conférence B	UAMOB	Président
Mr OUKAS	Maître assistant B	UAMOB	Encadreur
Mr Yahyaoui	Maître assistant B	UAMOB	Examineur
Mr Leboukh	Maître assistant B	UAMOB	Examineur

Année Universitaire 2018/2019

Remerciements

*Tout d'abord, nous tenons à remercier Allah, le clément et le miséricordieux
De nous avoir donné la force et le courage de mener à bien ce travail*

*Nous voudrions exprimer nos remerciements à notre encadreur **OUKAS Nouredine**, pour ses orientations, ses encouragements, sa disponibilité et ses précieux conseils qui nous ont permis de mener à bien ce travail.*

Nous tenons à remercier les membres du jury qui ont bien voulu accepter de juger notre modeste travail.

Nos remerciements s'adressent également à tous les enseignants du département informatique ayant participé d'une manière ou d'une autre à notre formation master ;

A la fin nos remerciements les plus sincères à toutes les personnes qui ont contribué de près ou de loin à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année Universitaire...

Dédicaces

Ce modeste travail est dédié à :

A celle qui a été toujours la source de grande affection.....ma MERE

*A celui qui a été toujours la source d'inspiration, de courage a tout le
long de mes études.....mon PERE*

A tous mes frères et sœurs

A toutes La famille (SETTI) et mes proches.

A tous mes amis sans exception.

*A toute la promotion informatique 2018/2019 que je leurs souhaite un
bon avenir.*

*A tous ceux qui m'ont aidé de près ou de loin à la réalisation de ce
modeste travail*

Rabah Setti

Dédicaces

Tous les mots ne sauraient exprimer la gratitude, l'amour, le respect, la reconnaissance, c'est tout simplement que : Je dédie cette mémoire à :

Mes très chers parents : Aucun hommage ne pourrait être à la hauteur de l'amour Dont ils ne cessent de me combler. Que dieu leur procure bonne santé et longue vie.

Tous mes proches de la famille Belkacem, et plus Particulièrement, mes sœurs et mes frères tous à son nom.

Tous mes chers amis et mes collègues. Tous ceux qui m'ont donné le soutien pour réaliser ce travail surtout.

Pour tout ce que vous avez fait pour moi et pour tout l'amour dont vous m'avez entouré.

Tous mes chers enseignants qui m'ont enseigné.

Belkacem Amir

Résumé

Les développements récents des technologies micro-capteurs et des communications sans fils ont engendré l'apparition des réseaux de capteurs sans fils (WSN : Wireless Sensor Networks). Un tel réseau est constitué d'un grand nombre de dispositifs physiques appelés nœuds de capteurs capables de relever, de traiter et de transmettre des informations de l'environnement, dans lequel ils sont déployés, à un ou plusieurs points de collecte. L'un des grands problèmes de ce type de réseaux est l'énergie des capteurs qui est limitée, cette contrainte doit être prise en compte afin d'allonger la durée de vie du réseau.

Parmi les solutions est d'introduire un mécanisme de veille pour conserver l'énergie. D'autre solution consiste à récupérer l'énergie depuis l'environnement (énergie solaire, vibrations, vent etc.) pour charger la batterie du capteur. Dans ce mémoire, nous proposons une modélisation des réseaux de capteurs sans fil par le formalisme des réseaux de Petri afin d'évaluer les performances des WSNs en prenant en considération le mécanisme de veille et la récupération de l'énergie.

Mots clés : : WSN, Réseaux de Petri, Modélisation, mécanisme de veille, Récupération d'énergie.

Abstract

Recent developments in micro-sensor technologies and wireless communications have led to the emergence of Wireless Sensor Networks (WSN). Such a network consists of a large number of physical devices called sensor nodes capable of picking up, processing and transmitting information from the environment in which they are deployed to one or more collection points (sink). One of the major problems of this type of network is the limited energy of the sensors, this constraint must be taken into account in order to extend the life of the network.

One of solutions is to introduce a sleeping mechanism to conserve energy. Another solution is to harvest the energy from the environment (solar, vibration, wind and others.) to charge the battery of the sensor. In this manuscript, we propose a modeling of the wireless sensor networks by the Petri nets formalism in order to evaluate the performances of these networks by taking into account the sleeping mechanism and the energy harvesting.

Key words: WSN, Petri nets, Modeling, sleeping Mechanism, Energy Harvesting.

Table des matières

I.	Introduction générale.....	1
II.	Généralité sur les Réseaux de Capteurs sans Fils.....	3
II.1.	Introduction	3
II.2.	Nœud capteur.....	3
II.2.1.	Architecture d'un nœud capteur.....	4
II.2.2.	Modèles de capteurs.....	5
II.3.	Réseaux de capteurs sans fil	6
II.3.1.	Définition d'un réseau de capteur sans fil.....	6
II.3.2.	Architecture d'un réseau de capteur sans fil	6
II.3.3.	Caractéristiques des réseaux de capteurs sans fil.....	7
II.3.4.	Contraintes de conception des RCSF :.....	8
II.3.5.	Classification des réseaux de capteurs :.....	11
II.3.6.	Technologies des réseaux de capteurs.....	12
II.3.6.1.	Bluetooth.....	12
II.3.6.2.	Zegbee	12
II.3.6.3.	UWB.....	12
II.3.6.4.	Infrarouge.....	12
II.3.7.	Domaines d'applications des réseaux de capteurs sans fil :	13
II.3.7.1.	Applications militaires	13
II.3.7.2.	Applications médicales.....	14
II.3.7.3.	Applications domestiques	14
II.3.7.4.	Applications environnementales.....	15
II.3.7.5.	Applications commerciales.....	15
II.4.	Conclusion :	16
III.	Consommation d'énergie dans les RCSFs	17
III.1.	Introduction	17
III.2.	Problème de l'énergie dans les RCSFs.....	17
III.3.	Facteurs intervenants dans la consommation d'énergie.....	19
III.3.1.	Technologie de communication	19
III.3.2.	Etat du module radio.....	19
III.3.3.	Accès au medium de transmission	19
III.3.3.1.	Retransmission	19

III.3.3.2. Ecoute active	20
III.3.3.3. Surécoute	20
III.3.3.4. Surcharge.....	20
III.3.3.5. Surémission	20
III.3.3.6. Taille des paquets.....	20
III.3.4. Modèle de propagation radio	20
III.4. Techniques de récupération d'énergie.....	21
III.5. La récolte d'énergie.....	21
III.5.1. Définition de la récolte d'énergie.....	21
III.5.2. Sources d'énergie	22
III.5.3. Techniques de la récolte d'énergie	22
III.5.4. Avantages	25
III.6. Conclusion	26
IV. Les réseaux de Petri	27
IV.1. Introduction	27
IV.2. Qualités et faiblesses des réseaux de Pétri	28
IV.3. Définition des RdPs.....	29
IV.4. L'aspect structurel	30
IV.4.1. Représentation d'un réseau de Pétri.....	30
IV.4.1.1. Représentation graphique.....	30
IV.4.1.2. Représentation matricielle	30
IV.4.1.3. Représentation d'un RdP marqué	31
IV.5. L'aspect comportemental.....	32
IV.5.1. L'état dans un réseau de Pétri.....	32
IV.5.2. Franchissement d'une transition.....	32
IV.5.3. L'exécution d'un réseau de Petri	33
IV.5.3.1. Exécution séquentielle	33
IV.6. Réseaux particuliers	35
IV.6.1. Graphe d'états	35
IV.6.2. Les réseaux sans conflits	35
IV.6.3. Les réseaux purs	36
IV.7. Propriétés des RdP	36
IV.7.1. Réseau K-borné	36
IV.7.2. Réseau vivant.....	37

IV.7.3.	Réseau réinitialisable.....	38
IV.8.	Les méthodes d'analyse des RdP.....	38
IV.8.1.	Méthode d'arbre de couverture	38
IV.8.2.	Approche d'équations matricielles.....	39
IV.8.3.	Technique de réduction et de décomposition	39
IV.9.	Réseaux de Pétri stochastiques généralisés (RdPSG).....	40
IV.9.1.	Evolution d'un RDPSG.....	40
IV.9.2.	Représentation graphique d'un RDPSG.....	41
IV.9.3.	Propriétés des RDPSG[44]	41
IV.9.4.	Analyse quantitative.....	41
IV.9.4.1.	Condition d'ergodicité.....	42
IV.9.4.2.	Evaluation des performances d'un RDPS	42
IV.9.4.3.	Calcul du générateur infinitésimal	42
IV.9.4.4.	Calcul du vecteur des probabilités stationnaires	43
IV.9.5.	Critères de performances	43
IV.9.5.1.	Fréquence moyenne de franchissement d'une transition ti	43
IV.9.5.2.	Nombre moyen de marques dans une place p	43
IV.9.5.3.	Temps moyen de séjour des marques dans une place p	43
IV.9.5.4.	Probabilité d'un événement A défini à travers une condition.....	44
IV.10.	Outils de modélisation des RDPS.....	44
IV.11.	Conclusion	45
V.	modélisation et Analyse d'un WSN avec les Réseaux de Pétri	46
V.1.	Introduction	46
V.2.	Un modèle GSPN pour WSNs	47
V.1.	Formules de Performance.....	49
V.2.	Conclusion	51
VI.	Application et étude expérimentale.....	52
VI.1.	Introduction	52
VI.2.	Description du logiciel <i>TimeNet</i>	52
VI.2.1.	Historique du logiciel <i>TimeNet</i>	52
VI.2.2.	Fonctionnalités du <i>TimeNet</i>	53
VI.2.1.	Interface graphique du <i>TimeNet</i>	53
VI.3.	Mesure de performances	54
VI.3.1.	Analyse qualitative du <i>TimeNet</i>	55

VI.3.2. Les méthodes d'analyse du <i>TimeNet</i>	57
VI.3.3. Analyse stationnaire	58
VI.3.4. Analyse numérique des résultats	59
VI.4. Conclusion	62
VII. Conclusion générale et perspectives.....	63
Bibliographie.....	64

Liste des figures

Figure II-1 Exemple des nœuds capteurs	3
Figure II-2 Architecteur matériel d'un capteur	4
Figure II-3 Architecteur d'un réseau de capteurs	6
Figure II-4 Pile protocolaire d'un réseau de capteurs.....	10
Figure II-5 Les RCSFs dans le domaine militaire.....	13
Figure II-6 Les RCSFs pour les applications sanitaires.....	14
Figure II-7 Les RCSFs pour les applications domestiques.....	15
Figure II-8 Les RCSFs pour la surveillance des applications environnementales.....	15
Figure III-1 La Surécoute dans une transmission.....	20
Figure III-2 Techniques de conservation d'énergie.....	21
Figure III-3 Récolteur d'énergie d'oreille.....	24
Figure III-4 Récolteur d'énergie du cœur	24
Figure III-5 Récolteur d'énergie à partir des mouvements de transpiration.....	25
Figure IV-1 Réseau de Petri simple.....	30
Figure IV-2 Réseau de Petri marqué.....	31
Figure IV-3 Graphe de Marquage.....	34
Figure IV-4 Graphe d'états ou pas.....	35
Figure IV-5 Conflit ou pas.....	35
Figure IV-6 RDP impur.....	36
Figure IV-7 RdP non borné.....	36
Figure IV-8 RdP vivant.....	37
Figure IV-9 Réseau réinitialisable.....	38
Figure IV-10 RDP et leur d'arbre de couverture	39
Figure V-1 Modèle GSPN du réseau de capteur sans fil	47
Figure VI-1 L'interface graphique du TimeNet sous Windows.....	54
Figure VI-2 Résultat de l'estimation de la taille de l'espace d'états	56
Figure VI-3 Résultat de l'exécution de la fonction <i>Traps</i>	56
Figure VI-4 Résultat de l'exécution de la fonction <i>Siphons</i>	57
Figure VI-5 Résultat de l'exécution de la fonction <i>Check Structure</i>	57
Figure VI-6 Menu d'évaluation du <i>TimeNet</i>	58
Figure VI-7 Fenêtre permet une analyse stationnaire.....	58
Figure VI-8 Fenêtre permet une exécution expérimentale	59
Figure VI-9 l'influence du taux_harvesting sur la moyenne charge de la batterie.....	59
Figure VI-10 le temps de réponse moyen par rapport au taux_sleep.....	60
Figure VI-11le temps de réponse moyen par rapport au taux_harvesting.....	60
Figure VI-12l'influence du taux_sleep sur la moyenne charge de la batterie.....	61
Figure VI-13temps de réponse moyen par rapport au nombre_msg N.....	61
Figure VI-14 le temps de réponse moyen par rapport au capacité de batterie C.....	62

Liste des tables

Tableau II-1 Les différents types de capteurs	5
Tableau II-2 Technologies sans fil et leurs caractéristiques.	13
Tableau IV-1 Les outils de modélisation des réseaux de Petri.....	45
Tableau V-1 Description des transitions temporisées.....	48
Tableau V-2 Description des Places	48

Introduction générale

Dans les premiers âges de l'informatique, la difficulté pour les chercheurs était d'adapter les différents algorithmes aux contraintes matérielles imposées par les ordinateurs (puissance de calcul, mémoire limitée, etc.). Même si ces difficultés n'ont pas disparu, leurs impacts tendent à diminuer tant la puissance des ordinateurs s'accroît à une vitesse surprenante. Il faut des applications vie à ces limites imposées par le matériel. Toutefois, les avancées technologiques conduisent à une nouvelle évolution du paradigme. Incontestablement, ce début de vingt et unième siècle est placé sous le signe de la communication. Après le phénomène Internet, la démocratisation des technologies sans fil révolutionne les moyens de communication avec notamment l'apparition de réseaux spontanés ou réseaux ad hoc. L'hétérogénéité de ces réseaux et l'absence d'infrastructure accroissent leurs intérêts et ouvrent de nouvelles perspectives avec, par exemple, l'émergence de réseaux mobiles.

Autre événement majeur de l'informatique, la décomposition et la répartition des calculs sur plusieurs machines a permis d'outre passer les limites des processeurs. En effet, pourquoi se limiter à une seule et même unité de calcul ? Dès lors qu'un calcul peut se décomposer, sa résolution sera d'autant plus rapide que le nombre d'unités de calcul sera élevé.

Enfin, indépendamment de ces avancées technologiques et algorithmiques, il existe une tendance dans le domaine des microsystèmes électroniques qui s'est accentuée ces dernières années : il s'agit de la miniaturisation. C'est dans ce contexte qu'apparaît une nouvelle génération d'appareils : les capteurs communicant. Ils sont généralement de petite taille, dotés d'une unité de calcul et capables de communiquer entre eux. On parlera alors de réseaux de capteurs sans fil (RCSFs). Les fonctionnalités des capteurs étant limitées de par leurs faibles ressources, les applications doivent être adaptées à leurs caractéristiques. L'émergence de ce nouveau champ d'étude va pousser les informaticiens à retourner aux sources de leur histoire. La tâche première d'un capteur est de détecter un événement (par exemple, un changement de température, des mouvements, des vibrations, ...). Il est donc capable de récolter des données relatives à son environnement, de les traiter, puis, si nécessaire, de les communiquer à des capteurs voisins via un médium sans fil. Le déploiement de ce type d'appareils forme alors un réseau qui peut être utilisé dans des domaines militaires (par exemple, le suivi de déplacement des troupes ennemies), civils (la détection de feux de forêt), médicaux (le suivi des patients), animaliers (l'étude des migrations d'espèces), etc. Dans plusieurs exemples, les capteurs sont mobiles. Il faut donc distinguer deux types de réseaux : les réseaux statiques et les réseaux mobiles. Si leurs perspectives d'utilisation sont claires et attrayantes, les problématiques qu'engendrent ces réseaux n'en sont pas moins nombreuses. A priori, ils ne dépendent d'aucune infrastructure et les capteurs n'ont aucune information relative au réseau auquel ils appartiennent. De plus, étant construits

de façon ad hoc, ces réseaux doivent être auto-organisant. Parmi les problèmes cruciaux, deux d'entre eux peuvent être cités :

- Celui de l'énergie : Les capteurs ont des capacités limitées d'énergie stockées dans des petites batteries.
- Celui du routage : il consiste à acheminer un message d'un capteur vers un autre. Souvent, les réseaux contiennent une station de base chargée de collecter l'ensemble des informations perçues par les capteurs. Il s'agit alors de transmettre ces informations point à point vers cette station de base.

Bien d'autres problèmes tels que l'adressage ou la diffusion sont liés à ce type de réseau. Chaque méthode proposée doit assurer l'auto-stabilisation du réseau en garantissant la convergence vers une solution stable.

L'énergie des capteurs étant limitée, cette contrainte doit être prise en compte afin d'allonger la durée de vie du réseau. Pour cela, nous introduisons dans ce mémoire un mécanisme de veille pour conserver l'énergie et la récupération d'énergie de puis l'environnement pour recharger les batteries. Nous utilisons le formalisme des Réseaux de Pétri pour modéliser, analyser et évaluer les performances de ce type de réseaux.

La suite de ce document est divisée en 5 chapitres comme suit :

Le chapitre 1 : Présente une description générale des réseaux de capteurs sans fil ainsi que leurs caractéristiques, contraintes et spécificités.

Le chapitre 2 : Nous présentons ici la consommation d'énergie dans les réseaux de capteur sans fil.

Le chapitre 3 : Donne une présentation générale des réseaux de pétri.

Le chapitre 4 : Présente la modélisation d'un WSN avec les réseaux de pétri.

Le chapitre 5 : Montre une présentation de l'analyse numérique et l'évaluation de performance des WSNs.

A la fin de ce mémoire, une conclusion générale est donnée pour résumer les apports essentiels de notre travail et pour donner des perspectives.

Chapitre 1

Généralité sur les Réseaux de Capteurs sans Fils

II.1. Introduction

Un réseau de capteurs est un ensemble de nœuds déployés dans une zone de capture pour mesurer des grandeurs physiques telles que : la température, la pression, la vibration, etc. Les données collectées par ces capteurs sont envoyées à une station de base et acheminées par cette dernière vers l'utilisateur final via un autre réseau de communication. Ce type de réseau est utilisé par plusieurs applications militaires et civiles telles que la santé, la surveillance, l'industrie, le transport, etc. Ces applications ont des exigences plus ou moins différents, particulièrement, en termes de sécurité et conservation d'énergie.

Dans ce chapitre, nous présentons l'architecture et les modèles d'un nœud capteur. Introduit les réseaux de capteurs, en particulier, leur architecture et caractéristiques, les facteurs pris en compte dans la conception de ce genre de réseau, la classification ainsi que les domaines d'applications. Le chapitre s'achève par une conclusion.

II.2. Nœud capteur

Un capteur est un petit appareil, comme illustré dans la (Figure II-1) autonome capable d'effectuer de simples mesures sur son environnement immédiat, telles que la température, les vibrations et la pression. Cet appareil est augmenté de capacités de calcul et de communication ainsi que de batteries lui conférant une autonomie.



Figure II-1 Exemple des nœuds capteurs

II.2.1. Architecture d'un nœud capteur

L'architecture d'un capteur comprend deux parties : matériel et système d'exploitation embarqué.

Un nœud capteur est composé de quatre unités [1] ou sous-systèmes principaux (Figure II-2).

Il s'agit des sous-systèmes de traitement, de communication, de capture ou de perception, et de contrôle d'énergie[1-3]

Il peut contenir également un ensemble supplémentaire de modules selon le domaine d'application, comme le système mobilisateur chargé de déplacer le micro-capteur en cas de nécessité, un générateur d'énergie[1, 2], ainsi qu'un système de localisation GPS (Global Positioning System) [4].

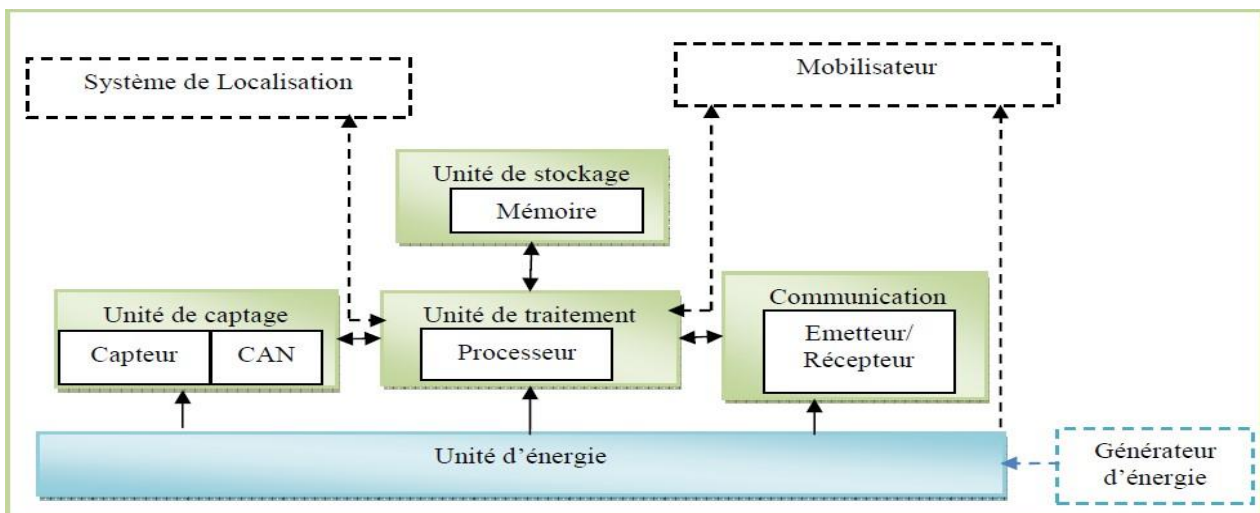


Figure II-2 Architecture matérielle d'un capteur[4]

- ❖ **Unité de capture** : est composée de deux sous-unités : un capteur et un convertisseur Analogique/Numérique (ADC pour Analog to Digital Converter)[5]. Le capteur permet

De relier le nœud avec son environnement extérieur, il est responsable de fournir des signaux analogiques, basé sur les phénomènes observés à l'ADC. Ce dernier transforme ces signaux en un signal numérique compréhensible par l'unité de traitement pour pouvoir l'analyser.

- ❖ **Unité de traitement** : est à la charge du contrôle des capteurs, la gestion des procédures et protocoles de communication[1, 6]. De plus le microcontrôleur ou le microprocesseur embarqué est associé généralement à une unité de stockage, nécessaire à l'implantation et à l'exécution d'un programme qui peut être un système d'exploitation spécialement conçu pour les micro-capteurs (TinyOS par exemple).

- ❖ **Unité de communication** : responsable d'effectuer toutes les émissions et réceptions de données sur un médium sans fil, afin de permettre l'échange d'informations entre le nœud capteur et son environnement extérieur. En fonction du domaine d'application, trois modes de communication sans fil peuvent être utilisés : optique (laser), infrarouge, radiofréquence (RF Radio Frequency)[3, 7, 8]
- ❖ **Unité d'énergie** : c'est la batterie qui est généralement, ni rechargeable ni remplaçable. Cette capacité d'énergie limitée au niveau des capteurs, représente une contrainte cruciale lors de la conception de protocoles pour les réseaux de capteurs. Par ailleurs, les unités d'énergie rechargeable : supportées par des photopiles (cellules solaire)[1] permettant de convertir l'énergie lumineuse en courant électrique et d'étendre la durée de vie de la batterie.

II.2.2. Modèles de capteurs

Les capteurs sont disponibles en différents modèles et dépendants de l'application à laquelle ils sont destinés. Il existe plusieurs fabricants de capteurs parmi lesquels on trouve : Imote IV, Art of Technology et Crossbow[9].

Les différents capteurs : MICA2, Telos B, MICAz, Imote2, IRIS, etc.

Les différents composants de chaque modèle ainsi que leurs caractéristiques sont décrit dans le Tableau II-1.

	Modèle	Miro-Contrôleur	Type Radio	Radio (kbps)	RAM	Flash	EEPROM	Batterie
Crossbow Technology	MICA2	Atmega 128L MPR400(8-bit)	ChipCon CC1000	38	4 KB	128 KB	4 KB	2xAA
	MICAz	Atmega 128L MPR2400(8-bit)	ChipCon CC2420	250	4 KB	128 KB	4 KB	2xAA
	TelosB	TI MSP 430 (16-bit)	ChipCon CC2420	250	10 KB	48 KB	16 KB	2xAA
	Imote2	Intel PXA271 (32-bit)	ChipCon CC2420	250	32 MB	32 KB		3xAA
Shockfish SA	Tanynode	TI MSP 430 (16-bit)	Semtech XE1205	153	10 KB	48 KB	16 KB	2/3xAA

Tableau II-1 Les différents types de capteurs [10].

II.3. Réseaux de capteurs sans fil

Les réseaux de capteurs sans fil sont l'une des technologies de la nouvelle génération de réseaux informatiques et télécommunications.

II.3.1. Définition d'un réseau de capteur sans fil

Un réseau de capteurs sans-fil est considéré comme un type spécial des réseaux ad hoc ou l'infrastructure fixe de communication et d'administration centralisée sont absentes et les nœuds jouent, à la fois, le rôle des hôtes et des routeurs. Les nœuds capteurs sont des capteurs intelligents (smart sensor), capables d'accomplir trois tâches complémentaires : le prélèvement d'une grandeur physique, le traitement éventuel de cette information et la communication avec d'autres capteurs. L'ensemble de ces capteurs déployés de manière aléatoire pour une application, forme un réseau de capteurs[11].

II.3.2. Architecture d'un réseau de capteur sans fil

Les nœuds capteurs sont souvent dispersés dans une zone géographique, appelée champs de capture qui définit la zone d'intérêt pour le phénomène observé. Les données captées sont acheminées au point de collecte comme montré dans la(Figure II-3)[12]. La station de base ou le nœud puits constitue une passerelle entre le réseau de capteurs et l'utilisateur final. Ainsi, l'utilisateur peut surveiller et commander l'environnement à distance via l'internet ou un satellite, en adressant des requêtes précisant le type de données requises à collecter aux autres nœuds du réseau via le nœud puits [8].

Les stations de base possèdent beaucoup plus de capacités que les capteurs tant au niveau de la mémoire que de la vitesse de traitement ou des réserves en énergie[1].

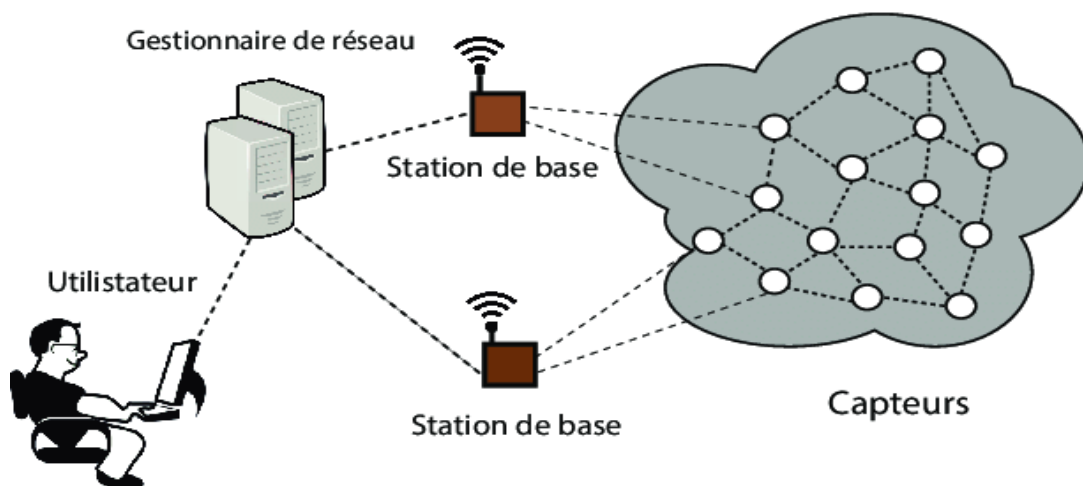


Figure II-3 Architecture d'un réseau de capteurs[8]

II.3.3. Caractéristiques des réseaux de capteurs sans fil

Les réseaux de capteurs sont caractérisés par :

Forte densité des nœuds : les RCSFs peuvent contenir des centaines voire des milliers de nœuds capteurs. La densité de nœud peut être aussi haute que 20 nœuds/m³[13].

Ressources limitées : les capteurs sont des objets limités en termes de : capacité mémoire, puissance de traitement, bande passante et énergie.

Contrainte d'énergie : dans plusieurs applications, les nœuds capteurs sont déployés dans des environnements hostiles. Dans ce cas, la durée de vie du capteur dépend de la vie de sa batterie, ce qui exige la minimisation des dépenses énergétiques. De ce fait, l'énergie est une contrainte clé dans les réseaux de capteurs[14, 15].

Déploiement de nœuds : il existe, en effet, plusieurs stratégies pour déployer un réseau de capteurs dans la zone à surveiller. Le choix d'une de ces stratégies dépend de l'objectif du réseau. Le déploiement peut être déterministe ou aléatoire. Dans la première stratégie, les capteurs sont placés manuellement et les données sont conduites via des chemins prédéterminés, mais cette solution devient impossible dès qu'on considère un très grand nombre de capteurs. En revanche, avec une approche aléatoire, les capteurs sont dispersés (exemple : lâchés d'un avion). Enfin, un mélange de ces deux techniques peut être envisagé[8, 14].

Connectivité : la densité élevée de nœud dans les réseaux de capteurs exclue complètement leur isolement. Quoique, la connexion n'est pas toujours assurée à cause de la mobilité, le déploiement et les défaillances qui font varier la topologie du réseau[8, 12].

Couverture limitée : dans les réseaux de capteurs, chaque nœud obtient une certaine vue locale de l'environnement, limitée par sa portée et sa précision. La couverture d'une vaste surface déterminée est composée de l'union de nombreuses couvertures de petite taille[8, 12].

Auto organisation du réseau et la topologie dynamique : ceci peut être nécessaire, vu le grand nombre de nœuds de micro-capteurs et leur placement dans des endroits hostiles, où l'intervention humaine n'est pas faisable. D'ailleurs, les nœuds peuvent échouer (par manque d'énergie ou destruction physique), comme de nouveaux nœuds peuvent rejoindre le réseau. Par conséquent, le réseau doit être capable de s'auto-organiser et de se modifier périodiquement de sorte qu'il puisse continuer à fonctionner [16].

Sécurité physique limitée : à l'instar des réseaux ad hoc, les RCSFs sont plus touchés par le paramètre de sécurité que les réseaux filaires classiques [17, 18].

II.3.4. Contraintes de conception des RCSF :

Plusieurs contraintes interviennent dans la conception d'un réseau de capteurs sans fil :

Tolérance aux pannes : La défaillance ou le blocage des nœuds dans un réseau de capteurs peut être engendré par plusieurs causes, notamment l'épuisement d'énergie, l'endommagement physique ou les interférences liées à l'environnement. La propriété de tolérance aux pannes est définie par l'habilité du réseau à maintenir ses fonctionnalités sans interruptions provoquées par la panne des capteurs. Elle vise donc à minimiser l'influence de ces pannes sur la tâche globale du réseau[1, 16].

Facteur d'échelle (scalabilité) : La surveillance d'un phénomène peut nécessiter le déploiement d'un nombre de nœuds qui est de l'ordre de plusieurs milliers de capteurs. Les nouveaux schémas doivent pouvoir garantir un bon fonctionnement avec ce nombre de capteurs élevé[1, 16].

Coût de production : Le coût de production d'un seul micro-capteur est déterminant pour l'évaluation du coût global du réseau. Si ce dernier est supérieur à celui nécessaire pour le déploiement des capteurs classiques, l'utilisation de cette nouvelle technologie ne serait pas financièrement justifiée. Par conséquent, réduire le coût de production jusqu'à moins de 1 dollar par nœud est un objectif important pour la faisabilité de la solution des réseaux de capteurs sans fil[16].

Contraintes matérielles : La plupart des algorithmes de routage dans les réseaux de capteurs sans fil requièrent la connaissance de la localisation des nœuds avec une grande précision. Ceci exige l'intégration d'une unité consacrée au système de localisation dans un nœud capteur. La conception des nœuds capteurs peut aller jusqu'à prévoir un système de mobilisation du capteur pour le déplacer en cas de nécessité.

Toutes ces unités peuvent exiger leur intégration dans un boîtier de taille minimale inférieure à un centimètre cube, et avec un poids très léger[16].

Support de transmission : Dans un réseau de capteurs multi sauts, les nœuds capteurs sont liés par un support sans fil. Ces liens de communications peuvent être des liaisons radio, infrarouge ou optique [16].

Topologie du réseau : Les caractéristiques de déploiement aléatoire, le fonctionnement autonome, et la fréquence élevée de pannes rendent la maintenance de la topologie d'un réseau de capteurs complexe. En effet, plusieurs centaines de capteurs sont déployés avec une densité pouvant être supérieure à 20 nœuds/m³, ceci exige une bonne gestion de la maintenance de la topologie du réseau déployé. Nous examinons, dans ce qui suit, les différents problèmes liés aux topologies des réseaux de capteurs et leurs changements[16].

- I. **Le pré-déploiement** : Les capteurs peuvent être éparpillés (manuellement ou par avion) ou placés précisément (manuellement ou par robots).
- II. **Le post-déploiement** : A cause des pannes, des déplacements de capteurs ou d'obstacles mobiles. La topologie doit parfois être changée après le déploiement.
- III. **Le redéploiement** : Des nœuds peuvent être déployés dans un deuxième temps pour remplacer ceux qui tombés en panne. Là encore la topologie devra s'adapter [19].

Environnement : Les nœuds capteurs doivent être conçus d'une manière à résister aux différentes et sévères conditions de l'environnement : forte chaleur, pluie et humidité [1].

Médias de transmission : Dans un réseau de capteurs, les nœuds sont reliés par une architecture sans fil. Pour permettre des opérations sur ces réseaux dans le monde entier, le média de transmission doit être standardisé. On utilise le plus souvent l'infrarouge, le Bluetooth[15] et les communications radio ZigBee .

Système d'exploitation : TinyOS est parmi les systèmes d'exploitation open-source pour les réseaux de capteurs, développé par l'université américaine de BERKELEY. Le caractère open-source permet à ce système d'être régulièrement enrichie par une multitude d'utilisateurs. Sa conception a été entièrement réalisée en NesC, langage orienté composant syntaxiquement proche du C. Il respecte une architecture basée sur une association de composants, réduisant ainsi la taille du code nécessaire à sa mise en place. Cela s'inscrit dans le respect des contraintes de mémoires qu'observent les capteurs pourvus de ressources très limitées dues à leur miniaturisation. Pour autant, la bibliothèque de composants de TinyOS est particulièrement complète puisqu'on y retrouve des protocoles réseaux, des pilotes de capteurs et des outils d'acquisition de données.

Un programme s'exécutant sur TinyOS est constitué d'une sélection de composants systèmes et de composants développés spécifiquement pour l'application à laquelle il sera destiné (mesure de température, du taux d'humidité...). TinyOS s'appuie sur un fonctionnement évènementiel, c'est-à-dire qu'il ne devient actif qu'à l'apparition de certains évènements, par exemple l'arrivée d'un message radio. Le reste du temps, le capteur se trouve en état de veille, garantissant une durée de vie maximale connaissant les faibles ressources énergétiques des capteurs [1].

Sécurité physique limitée : Les réseaux de capteurs sans-fil sont plus touchés par le paramètre de sécurité que les réseaux filaires classiques. Cela se justifie par les contraintes et limitations physiques qui font que le contrôle des données transférées doit être minimisé[1].

Topologie de réseau : Le déploiement d'un grand nombre de nœuds nécessite une maintenance de la topologie. Cette maintenance consiste en trois phases : déploiement,

post-déploiement (les capteurs peuvent bouger, ne plus fonctionner...) et redéploiement de nœuds additionnels[20].

Connectivité : Un réseau de capteurs est dit connecté si et seulement si, il existe au moins une route entre chaque paire de nœuds [21]. La connectivité dépend essentiellement de l'existence des routes.

Elle est affectée par les changements de topologie dus à la mobilité, la défaillance des nœuds, attaques, etc. Ce qui a pour conséquences : la perte de liens, l'isolement des nœuds, le partitionnement du réseau, la mise à jours des routes (le routage), etc.

Un graphe G est dit k -connecté, s'il y a au moins k chemins disjoints entre deux nœuds quelconques. La connectivité est une mesure de tolérance aux fautes ou de diversité de chemin dans le réseau. Le 1-connectivité est une condition fondamentale pour que le réseau soit opérationnel.

Consommation d'énergie : Les nœuds capteurs sont alimentés par des sources énergétiques limitées. Par conséquent, l'énergie d'un capteur doit être conservée pour augmenter sa durée de vie et la durée de vie du réseau tout entier en conséquence.

En général, dans un réseau de capteurs, l'énergie est consommée en trois opérations principales : la capture, la communication (émission et réception) et le traitement des données. La majorité des travaux de recherche menés actuellement se concentrent sur le problème de conservation d'énergie afin de concevoir des algorithmes et protocoles spécifiques à ce genre de réseau qui consomment le minimum d'énergie [16].

Pile protocolaire : La pile protocolaire, utilisée par la station de base ainsi que tous les autres capteurs d'un réseau de capteurs (Figure II-4) comprend cinq couches : couche application, couche transport, couche réseau, couche liaison de données et couche physique. Ainsi que Trois plans : plan de gestion d'énergie, plan de gestion de mobilité et plan de gestion des tâches.

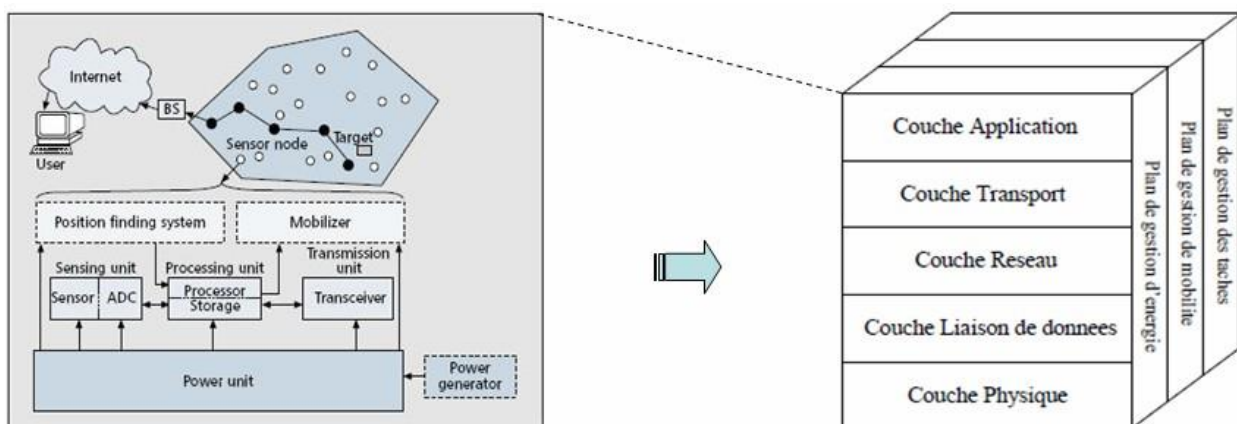


Figure II-4 Pile protocolaire d'un réseau de capteurs[16].

Couche physique : doit assurer des techniques d'émission, de réception et de modulation de données d'une manière robuste.

Couche liaison de données : cette couche doit garantir une faible consommation d'énergie en minimisant le taux de collision entre les données diffusées par les nœuds voisins. Elle est composée d'une couche de contrôle des liens logiques (LLC pour Logical Link Control) et une couche de contrôle d'accès au médium (MAC pour Médium Access Control).

Couche réseau : cette couche assure l'opération de routage qui consiste à découvrir les chemins des données.

Couche transport : La couche transport aide à gérer le flux de données si le réseau de capteurs l'exige. Elle permet de diviser les données issues de la couche application en segments pour les délivrer, ainsi elle réordonne et rassemble les segments venus de la couche réseau avant de les envoyer à la couche application.

Couche application : cette couche permet à l'utilisateur d'interagir avec le réseau de capteurs à travers des interfaces. Il s'agit donc du niveau le plus proche des utilisateurs, géré directement par logiciels.

Plan de gestion d'énergie : Les fonctions intégrées à ce niveau consistent à gérer l'énergie consommée par les capteurs. Dès lors, un capteur peut par exemple éteindre son interface de réception dès qu'il reçoit un message d'un nœud voisin afin d'éviter la réception des messages dupliqués. De plus, quand un nœud possède un niveau d'énergie faible, il peut diffuser un message aux autres capteurs pour ne pas participer aux tâches de routage, et conserver l'énergie restante aux fonctionnalités de capture[1].

Plan de gestion de mobilité : Ce niveau détecte et enregistre tous les mouvements des nœuds capteurs, d'une manière à leur permettre de garder continuellement une route vers l'utilisateur final, et le nœud peut garder trace de l'emplacement de ses voisins. En connaissant l'emplacement des voisins, les nœuds peuvent équilibrer l'usage de leur énergie.

Plan de gestion de tâche : Le niveau de gestion des tâches assure l'équilibrage et la distribution des tâches sur les différents nœuds du réseau, afin d'assurer un travail coopératif et efficace en matière de consommation d'énergie, et par conséquent, prolonger la durée de vie du réseau[1].

II.3.5. Classification des réseaux de capteurs :

Nous présentons une classification simple des réseaux de capteurs en fonction de leur mode de fonctionnement et du type de l'application cible :

- 1. Réseaux proactifs** : Les nœuds de ce réseau allument périodiquement leur capteurs et émetteurs, détectent l'environnement et transmettent les données d'intérêt. Ainsi, ils fournissent un aperçu de les paramètres pertinents à intervalles

réguliers. Ils sont bien adaptés aux applications nécessitant une surveillance périodique des données.[22].

2. **Le Réseaux réactifs** : Dans ce schéma, les nœuds réagissent immédiatement à une soudaine et des changements drastiques dans la valeur d'un attribut détecté. Comme tels, ils sont bien adaptés aux applications à temps critique[22].

II.3.6. Technologies des réseaux de capteurs[23]

Parmi les technologies radio utilisées pour permettre les communications entre capteurs, nous pouvons citer : Bluetooth (IEEE 802.15.1), UWB (ou IEEE 802.15.3) et Zigbee (IEEE 802.15.4) ...

II.3.6.1. Bluetooth

La technologie Bluetooth, dont Ericsson a initié le projet en 1994, a pour but principal de remplacer les câbles sur les petites distances. Elle est utilisée dans la plupart des téléphones portables comme interface de connexion pour accéder à un PC. Malheureusement, le grand défaut de cette technologie est sa trop grande consommation d'énergie. Donc elle ne peut pas être utilisée par des capteurs qui sont alimentés par une batterie et qui, idéalement devraient fonctionner durant plusieurs années [24].

II.3.6.2. Zigbee

Le standard Zigbee, combiné avec IEEE 802.15.4, offre des caractéristiques qui répondent encore mieux aux besoins des réseaux de capteurs. Zigbee offre des débits de données moindres, mais il consomme également nettement moins que Bluetooth. Un petit débit de données (250Kbit/s max) n'est pas handicapant pour un réseau de capteurs où les fréquences de transmission ne sont pas soutenues et conséquentes[24].

II.3.6.3. UWB

L'UWB peut être utilisé en tant que technique de communication sans fil, qui fournit des taux de transfert réseaux très élevés sur des distances relativement courtes et à faible puissance. De plus, l'atténuation du signal engendré par des obstacles est moindre qu'avec les systèmes radio à bande étroite conventionnels[25].

II.3.6.4. Infrarouge

Les ondes infrarouges sont utilisées pour la communication à courte distance, ainsi elles s'adaptent au mode de communication entre les nœuds d'un réseau de capteurs. Les ondes infrarouges sont préférées aux ondes radio, car elles n'interfèrent pas avec les autres signaux électromagnétiques.

Les émetteurs-récepteurs utilisant la technologie infrarouge ne sont pas chers, et ils sont faciles à fabriquer. Les composants matériels utilisant ce type de communication sont généralement conformes aux standards publiés par l'Infrared Data Association (IrDA).

Enfin, le tableau ci-dessous récapitule la différence entre les technologies ZigBee, Bluetooth et Wi-Fi.

Protocole	Zigbee	Bluetooth	Wifi
IEEE	802.15.4	802.15.1	802.11a/b/g
Besoin mémoire	4-32Ko	250 Ko	1 Mo
Energie nécessaire	Faible	Moyenne	Elevée
Nombre de nœuds	Non limité	7	32
Vitesse de transfert	250 Kb/s	720 Kb/s	11-54-108 Mb/s
Portée	100 m	1-100m	300m
Temps de démarrage	Court	Moyen	Long

Tableau II-2 Technologies sans fil et leurs caractéristiques.

II.3.7. Domaines d'applications des réseaux de capteurs sans fil :

L'intérêt des réseaux de capteurs est réellement vu à travers l'éventail très large des domaines d'applications.

Les applications des réseaux de capteurs peuvent être classées en cinq familles [26].

II.3.7.1. Applications militaires

Les réseaux de capteurs sans fil peuvent être considérés comme une partie intégrale des systèmes C4ISRT (Command, Control, Communications, Computing, Intelligence, Surveillance, Reconnaissance and Targeting) (Figure II-5).

Nous pouvons citer parmi les applications militaires : la surveillance du champ de bataille, examen des forces et terrains de l'ennemi, estimation des dégâts de la bataille, et la détection et reconnaissance des attaques nucléaires, biologiques et chimiques[27].



Figure II-5 Les RCSFs dans le domaine militaire[27].

II.3.7.2. Applications médicales

La surveillance des fonctions vitales de l'être humain serait possible grâce à des micro-capteurs qui pourront être avalés ou implantés sous la peau.

Actuellement, des micro-caméras qui peuvent être avalées existent. Elles sont capables, sans avoir recours à la chirurgie, de transmettre des images de l'intérieur d'un corps humain (Figure II-6).

L'utilisation des réseaux de capteurs dans le domaine de la médecine pourrait apporter une surveillance permanente des patients et une possibilité de collecter des informations physiologiques de meilleure qualité, facilitant ainsi le diagnostic de quelques maladies [27].

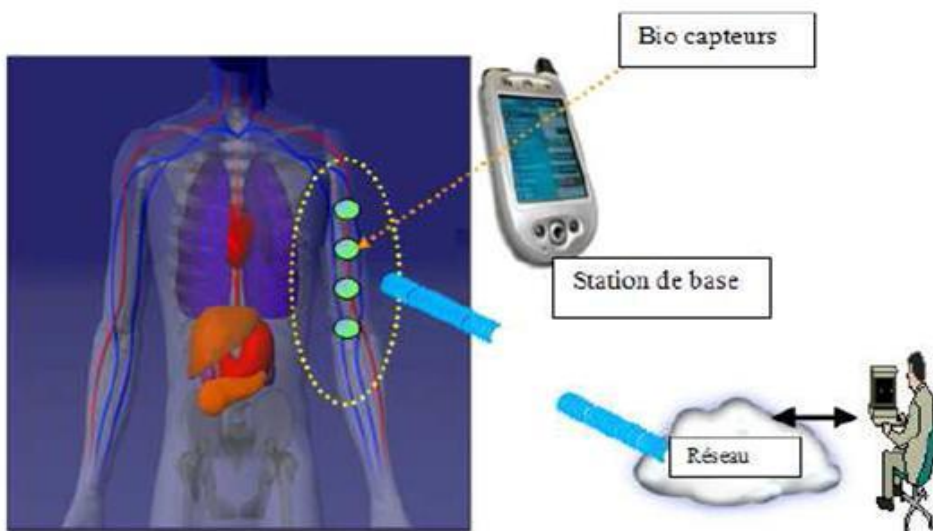


Figure II-6 Les RCSFs pour les applications sanitaires [27].

II.3.7.3. Applications domestiques

Comme la technologie progresse, les nœuds capteurs intelligents et actionneurs peuvent être embarqués dans des dispositifs, comme l'aspirateur, micro-onde, réfrigérateurs.

Ces capteurs dans les dispositifs domestiques peuvent interagir entre eux et avec un réseau externe par Internet ou satellite. Ils permettent à l'utilisateur final de gérer les dispositifs domestiques localement ou à distance plus facilement (Figure II-7) [27].

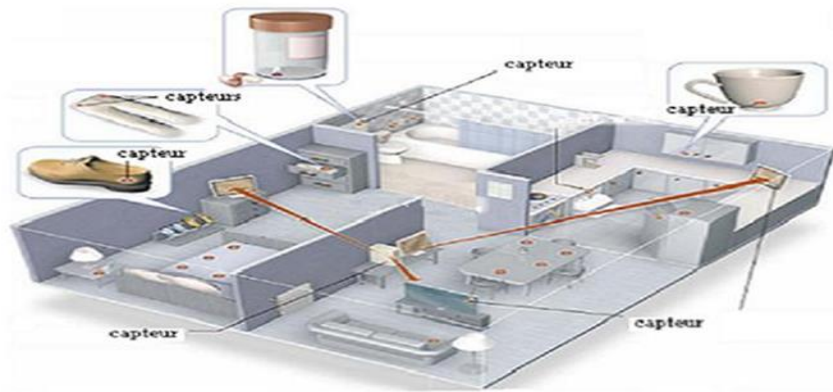


Figure II-7 Les RCSFs pour les applications domestiques[27].

II.3.7.4. Applications environnementales

Les réseaux de capteurs peuvent être utilisés dans les champs agricoles, sur les sites industriels, dans les centrales nucléaires, dans les pétroliers, dans les forêts ou pour la surveillance de l'environnement marin (Figure II-8).

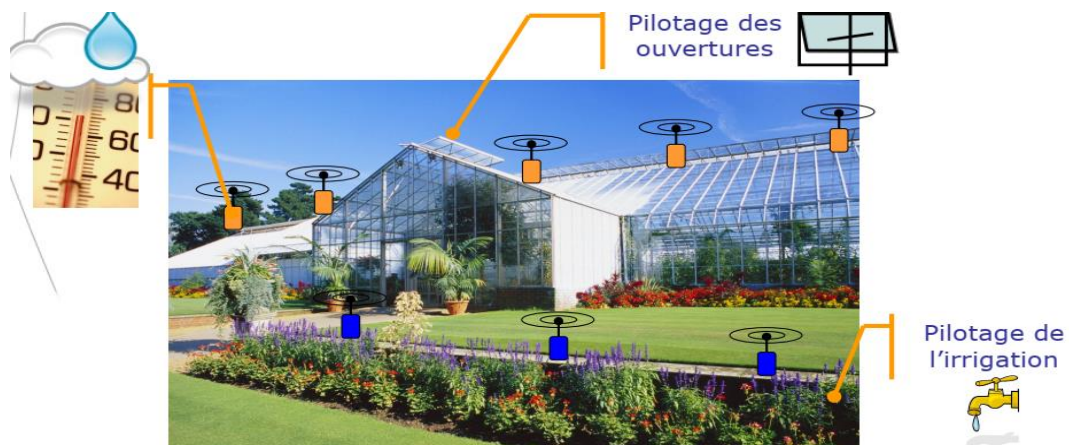


Figure II-8 Les RCSFs pour la surveillance des applications environnementales[27].

II.3.7.5. Applications commerciales

Pour les entreprises, les réseaux de capteurs permettront de suivre le procédé de production à partir des matières premières jusqu'au produit final livré [27].

II.4. Conclusion :

Un réseau de capteurs est une collection de capteurs déployés dans une zone de capture pour prélever des mesures physiques.

Dans ce chapitre, nous avons introduit ce type de réseau, en particulier, nous avons présenté l'architecture, les caractéristiques, les facteurs influant sur la conception des réseaux de capteurs, ainsi que les domaines d'applications de ce genre de réseau. Il a été constaté que la recherche dans les réseaux de capteurs est beaucoup plus orientée vers la conservation de l'énergie afin de prolonger la durée de vie du réseau.

Dans le chapitre suivant, nous allons mettre l'action sur la consommation d'énergie dans les réseaux de captures sans fils

Chapitre **2**

Consommation d'énergie dans les RCSFs

III.1. Introduction

Dans les réseaux de capteurs sans fil, la consommation d'énergie est très importante puisque généralement les capteurs sont déployés dans des zones inaccessibles. Ainsi, il est difficile voire impossible de remplacer les batteries après leur épuisement. De ce fait, la consommation d'énergie au niveau des capteurs a une grande influence sur la durée de vie du réseau.

Dans ce chapitre, nous allons étudier la consommation d'énergie dans ce type de réseaux sans fil. Pour cela, nous avons organisé notre travail comme suit : nous allons évoquer tout d'abord le problème d'énergie au sein d'un réseau de capteurs. Nous allons présenter, ensuite les différents facteurs intervenants dans la dissipation d'énergie des capteurs. Puis, nous allons montrer les différentes techniques proposées et créées afin de minimiser cette consommation.

III.2. Problème de l'énergie dans les RCSFs

La consommation d'énergie est un problème fondamental lorsque les capteurs sont déployés dans des zones inaccessibles ou encore déployés sur de grands espaces. Pour cela, les nœuds limités en termes d'énergie vont avoir un impact sur la durée de vie du réseau tout entier. Augmenter la durée de vie du réseau sans augmenter la capacité des batteries signifie réduire la dépense d'énergie des nœuds.

Pour le cas des applications de surveillance par exemple exigeant que le réseau de capteurs soit opérationnel pendant une très longue durée, économiser l'énergie est un véritable challenge.

La littérature scientifique traitant l'efficacité énergétique dans les réseaux de capteurs est abondante. Avant de présenter les techniques permettant de rendre les réseaux de capteurs sans fil moins gourmands en énergie, nous commençons par une analyse des principales sources de dissipation d'énergie.

L'énergie de capture est dissipée pour accomplir les tâches suivantes :

Échantillonnage, traitement de signal, conversion analogique/numérique et activation de la sonde de capture.

Cette énergie consommée au moment de la mesure varie suivant la nature du capteur. En effet, pour la capture des données scalaires simples, comme la température, vibrations, humidité ou la pression, quelques octets est largement suffisante pour leur codage (2 ou 3 octets). Tandis que pour une image il est généralement représenté sur plusieurs milliers d'octets (16384 octets pour une image de taille 128 × 128 codée sur 8 bits par pixel).

Malgré ces variétés, l'énergie d'acquisition représente un faible pourcentage de l'énergie totale consommée.

- **Energie de communication :**

L'énergie de communication est déterminée par la quantité des données à communiquer et la distance de transmission, ainsi que par les propriétés physiques du module radio.

Cette énergie se décline en deux parties : l'énergie de réception et l'énergie de l'émission. Le coût de réception est fixe mais le coût en émission dépend de la puissance d'émission du signal radio, de débit binaire et de la taille des données à transmettre.

Quand la puissance d'émission est élevée, le signal aura une grande portée et l'énergie consommée sera plus élevée.

Notons que cette partie d'énergie dépensée représente la portion la plus grande de l'énergie totale consommée par un nœud capteur. C'est pour cette raison que nous allons étudier par la suite les techniques de minimisation de la consommation de cette énergie [15].

- **Energie de traitement :**

L'énergie de traitement se divise en deux parties : l'énergie de commutation et l'énergie de fuite. L'énergie de commutation est déterminée par la tension d'alimentation et la capacité totale commutée au niveau logiciel (en exécutant un logiciel). Par contre, l'énergie de fuite correspond à l'énergie consommée lorsque l'unité de calcul n'effectue aucun traitement.

En général, l'énergie de traitement est faible par rapport à celle nécessaire pour la communication [15].

III.3. Facteurs intervenants dans la consommation d'énergie

La consommation d'énergie dépend de plusieurs facteurs qui sont expliqués ci-dessous :

III.3.1. Technologie de communication

L'unité de communication met en œuvre des protocoles de communication dépendant de la technologie utilisée (par exemple Wi-Fi (802.11), Bluetooth (802.15.1), ZigBee (802.15.4), UWB (802.15.3) pour les technologies sans fil) et ils n'ont pas le même coût d'énergie.

III.3.2. Etat du module radio

Le module radio est le composant du nœud capteur qui consomme le plus d'énergie, puisque c'est lui qui assure la communication entre les nœuds. On distingue quatre états des composants radio (transmetteur et récepteur) : actif, réception, transmission et sommeil[28].

- **Etat actif** : la radio est allumée, mais elle n'est pas employée. En d'autres termes, le nœud capteur n'est ni en train de recevoir ni de transmettre. Cet état provoque une perte de l'énergie suite à l'écoute inutile du canal de transmission.
- **Etat sommeil** : la radio est mise hors tension.
- **Etat transmission** : la radio transmet un paquet.
- **Etat réception** : la radio reçoit un paquet.

Il est aussi à noter que le passage fréquent de l'état actif à l'état sommeil peut avoir comme conséquence une consommation d'énergie plus importante que de laisser le module radio en mode actif. Ceci est dû à la puissance nécessaire pour la mise sous tension du module radio. Cette énergie est appelée l'énergie de transition. Il est ainsi souhaitable d'arrêter complètement la radio plutôt que de transiter dans le mode sommeil. Le changement d'état du module radio doit être géré par un protocole de la couche MAC.

III.3.3. Accès au medium de transmission

La couche MAC joue un rôle important pour la coordination entre les nœuds et la minimisation de la consommation d'énergie. Dans cette section, nous allons analyser les principales causes de consommation d'énergie au niveau de la couche MAC[29].

III.3.3.1. Retransmission

Les nœuds capteurs possèdent en général une seule antenne radio et partagent le même canal de transmission. Par ailleurs, la transmission simultanée des données provenant de plusieurs capteurs peut produire des collisions et ainsi une perte de l'information transmise.

La retransmission des paquets perdus peut engendrer une perte significative de l'énergie.

III.3.3.2. Ecoute active

L'écoute active (*idle listening*) du canal pour une éventuelle réception de paquet qui ne sera pas reçu peut engendrer une perte importante de la capacité des nœuds en énergie. Pour éviter ce problème, il faut basculer les nœuds dans le mode sommeil le plus longtemps possible.

III.3.3.3. Surécoute

Le phénomène de Surécoute (*overhearing*) se produit quand un nœud reçoit des paquets qui ne lui sont pas destinés (Figure III-1).

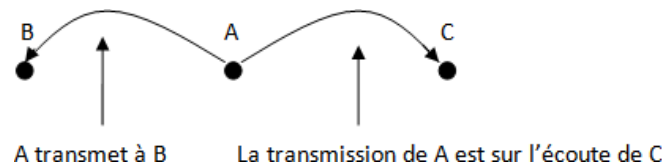


Figure III-1 La Surécoute dans une transmission[29].

La Surécoute conduit à une perte d'énergie additionnelle à cause de l'implication des autres capteurs dans la réception des données.

III.3.3.4. Surcharge

Plusieurs protocoles de la couche MAC fonctionnent par échange de messages de contrôle (*overhead*) pour assurer différentes fonctionnalités : signalisation, connectivité, établissement de plan d'accès et évitement de collisions. Tous ces messages nécessitent une énergie additionnelle.

III.3.3.5. Surémission

Le phénomène de surémission (*overemitting*) se produit quand un nœud capteur envoie les données à un destinataire qui n'est pas prêt à les recevoir. En effet, les messages envoyés sont considérés inutiles et consomment une énergie additionnelle.

III.3.3.6. Taille des paquets

La taille des messages échangés dans le réseau a un effet sur la consommation d'énergie des nœuds émetteurs et récepteurs. Ainsi, la taille des paquets ne doit être ni trop élevée ni trop faible. En effet, si elle est petite, le nombre de paquets de contrôle (acquiescement) générés augmente l'*overhead*. Dans le cas contraire, une grande puissance de transmission est nécessaire pour des paquets de grande taille.

III.3.4. Modèle de propagation radio

Le modèle de propagation représente une estimation de la puissance moyenne reçue du signal radio à une distance donnée d'un émetteur. La propagation du signal radio est généralement soumise à différents phénomènes : la réflexion, la diffraction et la dispersion par divers objets[30].

III.4. Techniques de récupération d'énergie

Après la description des principales causes de consommation d'énergie dans les RCSFs et la technique de veille utilisée pour minimiser la consommation d'énergie, nous présentons ici la technique de récupération d'énergie. La Figure III-2 résume les techniques de conservation d'énergie.

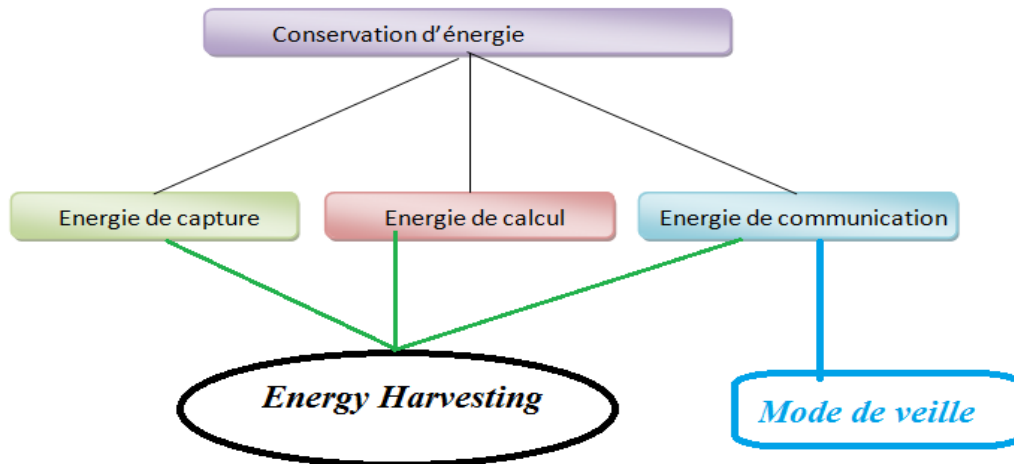


Figure III-2 Techniques de conservation d'énergie.

III.5. La récolte d'énergie

Comme il a été mentionné dans la section précédente, les RCSF forment une classe de réseaux sans fil qui doivent faire face à plusieurs défis comme la fiabilité et la robustesse dans des environnements difficiles. En outre, les dispositifs qui forment les RCSF devraient avoir une longue durée de vie et un faible coût, un compromis difficile à satisfaire. À ces termes, la technologie de la récolte d'énergie s'avère intéressante à utiliser comme source d'énergie potentielle. Cette section introduit le concept de la récolte d'énergie puis explicite ses principales techniques et son importance pour les RCSF.

Remarque

Les termes récupération, accumulation, récolte d'énergie et *energy harvesting* sont utilisés pour désigner le même concept.

III.5.1. Définition de la récolte d'énergie

La récolte d'énergie est la conversion de l'énergie ambiante présente dans l'environnement en une énergie électrique et son stockage dans des unités comme des batteries. L'énergie ambiante est naturelle, de nature non-électrique, et renouvelable. Toutefois, le principal avantage de la récolte d'énergie est d'éliminer la nécessité de mettre en place des fils ou de faire des visites fréquentes pour remplacer les batteries des dispositifs [31].

III.5.2.Sources d'énergie

En général, il existe différents types de sources d'énergie à partir desquelles il est possible de récolter l'énergie. Le rendement d'une source dépend essentiellement de l'emplacement du récolteur d'énergie vis à vis cette source. Le choix de la source dépend essentiellement de l'application en question. Par exemple à l'intérieur, l'éclairage artificiel peut fournir assez de puissance pour les applications de faible puissance, telles que les calculatrices. Le processus de la récolte d'énergie peut obtenir de l'énergie utilisable à partir de sources naturelles qui nous entourent dans l'environnement de tous les jours, telles que le mouvement physique, les ondes électromagnétiques, les rayons du soleil, le vent, les vagues de l'océan, les courants des rivières et les ondes sonores [31].

III.5.3.Techniques de la récolte d'énergie

Dans ce qui suit, quelques exemples des techniques de récolte d'énergie fréquemment utilisées sont présentés.

- **Conversion de l'énergie mécanique**

L'énergie mécanique peut être trouvée presque partout où les RCSF sont déployés, ce qui rend la conversion de l'énergie mécanique à partir des vibrations ambiantes en énergie électrique une approche intéressante pour l'alimentation des capteurs sans fil. La récolte d'énergie mécanique signifie la conversion de l'énergie mécanique en électricité en utilisant les vibrations, la pression, les moteurs à haute pression et la force [32].

- **Conversion de l'énergie photovoltaïque**

La récolte de l'énergie photovoltaïque est le processus de conversion des photons provenant des sources telles que la lumière solaire ou artificielle en électricité. À partir de l'énergie photovoltaïque, on peut produire des niveaux plus élevés d'électricité par rapport à d'autres techniques de récolte d'énergie. Ceci rend ces sources convenables aux systèmes de récolte d'énergie à grande échelle [32].

Cependant, la puissance générée et l'efficacité du système dépendent fortement de la disponibilité de la lumière et des conditions environnementales ainsi que des matériaux utilisés[31 , 32].

- **Conversion de l'énergie thermique**

La récolte d'énergie thermoélectrique repose sur la production de l'énergie électrique à partir de la différence de température. Cette technique consiste généralement en l'utilisation de générateurs thermoélectriques. Le récolteur d'énergie accumule l'énergie électrique lorsqu'une tension est créée par deux métaux joints à deux jonctions qui sont maintenues à des températures différentes. La différence de tension est proportionnelle à la différence de température entre les plaques chaudes et froides.

L'énergie est récoltée tant que la différence de température est maintenue [31]. La récolte d'énergie thermoélectrique fournit des niveaux d'énergie récoltés élevés.

- **Conversion de l'énergie chimique**

La récolte d'énergie chimique est le processus de conversion de l'oxygène et des substances endogènes en électricité par l'intermédiaire de réactions électrochimiques. En particulier, les cellules de biocarburants agissant comme enzymes actives et les catalyseurs peuvent être utilisés pour convertir l'énergie biochimique en énergie électrique. Parmi ceux-ci, le glucose est considéré comme la source la plus commune des combustibles utilisés. Même si la récolte d'énergie biochimique peut être supérieure à d'autres techniques de récolte d'énergie en termes de puissance de sortie continue, son rendement est fonction du type et de la disponibilité des piles à combustible [31].

- **Conversion de l'énergie acoustique**

La récolte de l'énergie acoustique est le processus consistant à convertir les ondes acoustiques de l'environnement en énergie électrique au moyen d'un résonateur. En règle générale, la récolte d'énergie acoustique est utilisée dans le cas des endroits éloignés ou isolés, ou des systèmes fermés où le câblage et les commutations électriques sont difficiles à utiliser. Cependant, l'efficacité de la puissance acoustique récoltée est faible et une telle énergie ne peut être récoltée que dans des environnements très bruyants [31].

- **Récolte de l'énergie à partir du corps humain**

Le corps humain constitue une source intéressante pour la production d'énergie à partir de la dissipation de chaleur, des rotations, l'application du poids du corps ainsi que la déformation élastique des tissus et autres accessoires. En effet, le corps humain brûle en moyenne environ $10,5 \text{ MJ}$ d'énergie par jour, ce qui est l'équivalent d'une dissipation de puissance moyenne de 121 W . Cela ouvre des possibilités de récolte d'énergie pour alimenter des dispositifs médicaux implantables ou mobiles qui pourraient être utilisés pendant une longue période ou être rechargés en permanence [31]. Dans ce qui suit, on cite quelques appareils de mouvement humain utilisés pour la récolte d'énergie.

Parmi les sources possibles, on trouve l'oreille. En effet, on peut récolter l'énergie en utilisant le potentiel électrique de la cochlée comme une batterie biologique qui est essentielle pour transformer les ondes de pression sonores en signaux électriques envoyés au cerveau. Les chercheurs ont mis au point une puce qui peut exploiter cette énergie électrique sans interférer avec une audition normale telle qu'illustré dans la Figure III-3.

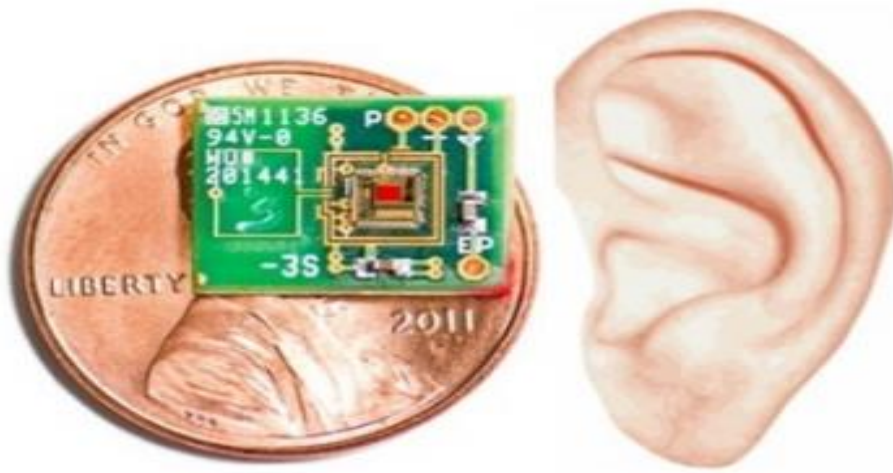


Figure III-3 Récolteur d'énergie d'oreille[31]

Des chercheurs français ont aussi développé un stimulateur cardiaque de faible puissance ($5\mu\text{W}$ au lieu de $25\mu\text{W}$ dans les stimulateurs actuels) alimenté par l'énergie mécanique de battements cardiaques d'un patient. L'objectif est d'éliminer le besoin de batteries, qui devaient être remplacé chirurgicalement tous les six à dix ans. Ce système est illustré par la Figure III-4.

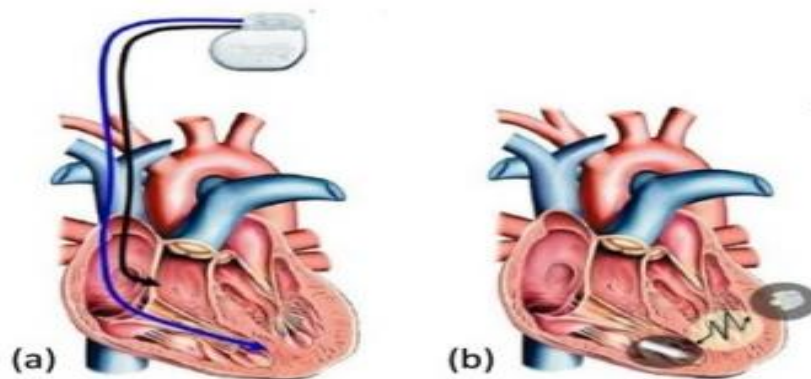


Figure III-4 Récolteur d'énergie du cœur[31]

João Paulo Lammoglia, un designer industriel basé à Londres, a créé un masque qui convertit l'énergie éolienne fournie par la respiration du porteur en électricité pour la recharge de petits appareils électroniques. À l'intérieur de l'unité, il y a des petites éoliennes qui font la conversion et l'énergie est transférée par l'intermédiaire d'un câble au petit appareil électronique. Ce masque est présenté à la Figure III-5.

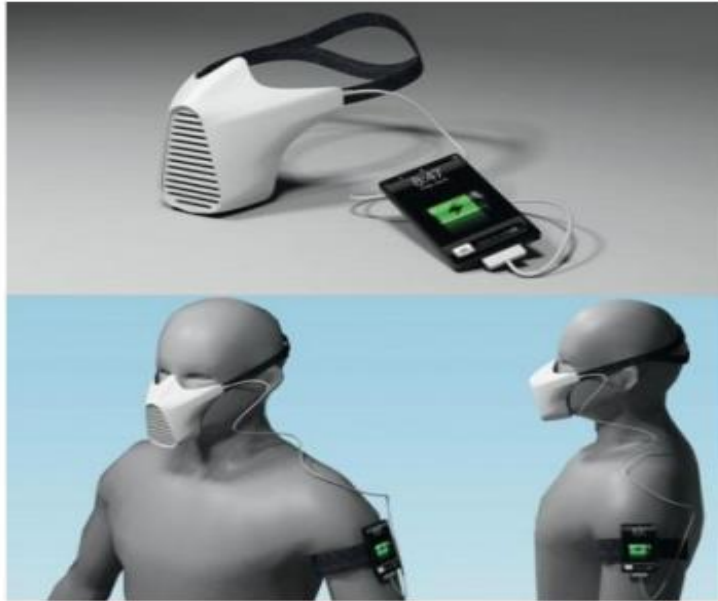


Figure III-5 Récolteur d'énergie à partir des mouvements de transpiration[31]

- **Conversion de l'énergie des signaux radio fréquence (RF)**

Il est possible de récolter l'énergie à partir des signaux radio fréquence (RF) qui se propagent dans l'air. Par exemple, à partir des signaux Wifi, les signaux de signalisation dans les réseaux cellulaires, etc. Cette technique de récolte d'énergie requiert une densité suffisante de signaux RF pour être effective.

III.5.4. Avantages

Nous pouvons utiliser la récolte d'énergie pour fournir de l'électricité pour les éléments aussi petits que les capteurs ou aussi grands que les satellites. Les principales raisons de son utilisation sont [31] :

- **Commodité** : Les consommateurs ne sont pas soucieux de changer ou de recharger les batteries pour les appareils tels que les ordinateurs portables ou les téléphones cellulaires ou d'autres appareils électroniques.
- **Sauvegarde d'énergie** : De tels dispositifs récolteurs d'énergie peuvent servir comme alternative aux sources d'énergie primaires. En d'autres termes, ils représentent un secours et constituent une solution aux coupures de courant. Par exemple, ceci est important pour les hôpitaux qui ont besoin d'énergie, surtout lors des pannes des sources primaires.
- **Pratiques commerciales** : Les coûts proposés aux consommateurs seraient réduits en termes d'emballage, de développement, d'élimination, de longévité et de réutilisation de certains produits. L'installation et la maintenance du produit aurait également un coût réduit. Un exemple est celui des coûts qui seraient enregistrés avec l'élimination des batteries et des chargeurs chimiques, ainsi que le vaste câblage exigé normalement dans l'utilisation des téléphones cellulaires.

III.6. Conclusion

Contrairement aux réseaux traditionnels qui se préoccupent de garantir une bonne qualité de service, les réseaux de capteurs sans fil doivent, en plus, prendre en compte la conservation d'énergie.

Ils doivent intégrer des mécanismes qui permettent aux utilisateurs de prolonger la durée de vie du réseau en entier, car chaque nœud est alimenté par une source d'énergie limitée et généralement irremplaçable. Dans un nœud capteur, l'énergie est consommée en assurant les fonctions suivantes : la capture, le calcul (traitement) et la communication. Cette dernière représente une grande portion de l'énergie totale consommée. De ce fait, nous avons présenté la technique de veille de récupération d'énergie pour résoudre le problème d'énergie dans les RCSFs.

Chapitre 3

Les réseaux de Petri

IV.1. Introduction

Les Réseaux de Petri représentent un outil mathématique puissant dans le domaine de la modélisation et de la vérification des systèmes. En plus de leur force d'analyse il offre une représentation graphique simple qui aide à la modélisation des systèmes complexes.

Le formalisme formel des Réseaux de Petri (RdP), adapté à la prise en compte des problèmes de concurrence, de synchronisme et de parallélisme, constitue un excellent outil de spécification fonctionnelle d'un problème et de mise en évidence des contraintes. Les propriétés mathématiques qui découlent de l'analyse des RdPs permettent une étude comportementale et structurelle essentielle à la validation d'une spécification. Les possibilités de simulation offertes par les outils informatiques supportant le formalisme contribuent également à cette validation.

En général, les méthodes de l'étude d'un système par RdP se composent de trois étapes : premièrement on écrit le système en termes de réseau, pour obtenir un modèle en réseau ; deuxièmement on analyse le modèle obtenu, pour en déduire des propriétés comme l'absence de blocage, existence d'une solution, etc. Finalement, on fait la révision des propriétés obtenues pour montrer si le système est bon. Le résultat de cette méthode nous indique une analyse qualitative du système. Elle constitue une approche très importante pour avoir une bonne évaluation des systèmes.

Nous introduirons dans ce chapitre la notion des réseaux de Pétri, nous présenterons par la suite quelques propriétés et extensions.

IV.2. Qualités et faiblesses des réseaux de Pétri

En tant que formalisme de spécification, les réseaux de Pétri présentent un certain nombre de qualités très importantes [33]:

- **Ils disposent d'une définition formelle** : ce caractère formel permet de produire des spécifications exemptes d'ambiguïté ; chaque construction des modèles possède une sémantique parfaitement définie.
- **Ils présentent un grand pouvoir d'expression** : les RDP sont notamment très bien adaptés à décrire des comportements complexes, réactifs ou concurrents.
- **Ils sont exécutable** : les modèles peuvent être interprétés par un programme construit à partir de la définition formelle de la notation, ce qui permet de simuler le fonctionnement du système en cours de spécification. Le modélisateur profite ainsi d'une vision dynamique du système qu'il spécifie pour approfondir la compréhension qu'il a de son comportement.
- Ils disposent de nombreuses techniques de vérification automatique des propriétés des modèles. Il est possible de rechercher des propriétés génériques telles que le caractère borné, vivant ou réinitialisable, ou des propriétés spécifiques telles que l'existence d'invariants.
- Ils disposent d'une représentation graphique attrayante, qui accroît la lisibilité et facilite la compréhension des modèles. Cette représentation graphique est également très utile lors de l'exécution interactive des modèles, servant alors de « débogueur » graphique.

Toutefois, malgré ces qualités, il semble que les réseaux de Pétri souffrent de certains reproches

- **Leur manque de structuration** : l'utilisation des RDPs produirait des modèles dont la taille croît rapidement avec la complexité du système, et qui deviennent rapidement impossibles à comprendre et à gérer.
- **Leur difficulté à prendre en compte l'aspect « structure de données »** : A l'origine, les RDPs étaient essentiellement destinés à décrire la structure de contrôle d'un système et son évolution dynamique. Les réseaux Place/Transition échouent en effet à décrire la structure des données manipulées par le système, leur fermeture à la modélisation de l'environnement extérieur et la manière dont ces données peuvent influencer le comportement dynamique.

Ces reproches ont certainement été légitimes à une époque, mais ils ont été entendus par la communauté RDPs, qui a développé des techniques de structuration des réseaux, et qui a traité le problème de la prise en compte des données en développant la théorie des réseaux de Petri de haut niveau.

IV.3. Définition des RdPs

Les réseaux de Pétri sont définis comme étant un formalisme qui permet la description et l'analyse du comportement des systèmes concurrents, introduit par *Carl Adem Petri* en 1962[34]:

Un réseau de Pétri (R) est un triple $R = (P, T, W)$ où P est l'ensemble des places (les places représentent les sites) et T l'ensemble des transitions (les transitions représentent les actions) tel que $P \cap T = \emptyset$ et W est la fonction définissant le poids porté par les arcs tel que :

$W: ((P \times T) \cup (T \times P)) \rightarrow \mathbb{N} = \{0, 1, 2, \dots\}$	Équation IV-1
---	----------------------

Le réseau R est fini si l'ensemble des places et des transitions est fini c.-à-d.

$P \cup T \in \mathbb{N}$ Un réseau $R = (P, T, W)$ est ordinaire si pour toute $(x, y) \in ((P \times T) \cup (T \times P))$: $W(x, y)$

Dans un réseau ordinaire la fonction W est remplacée par F où : $F \subseteq ((P \times T) \cup (T \times P))$

Tels que $(x, y) \in F \Leftrightarrow W(x, y) \neq 0$

Pour chaque $x \in P \cup T$:

- *x représente l'ensemble des entrées de x : ${}^*x = \{y \in P \cup T \mid W(y, x) \neq 0\}$
- x^* représente l'ensemble des sorties de x : $x^* = \{y \in P \cup T \mid W(x, y) \neq 0\}$

Remarque : si ${}^*x = \emptyset$, x est dite source, si $x^* = \emptyset$, x est dite puits.

➤ Les définitions concernant les **réseaux de Pétri** portées sur deux aspects :

- **Un aspect structurel** : Quelles sont les actions, quels sont les sites, quelles sont les conditions pour qu'une action soit possible et quelles sont les conséquences d'une action ?
- **Un aspect comportemental** : Comment représenter le fonctionnement d'un réseau de Pétri ? c.-à-d. ce qui se passe quand une action ou plusieurs actions sont exécutées.

IV.4. L'aspect structurel

IV.4.1. Représentation d'un réseau de Pétri

IV.4.1.1. Représentation graphique

L'un des aspects les plus agréables des réseaux de Pétri est qu'il est extrêmement aisé de les visualiser ; c.-à-d., donner une interprétation graphique à sa structure qui peut être représentée à travers un **graphe** bipartite fait de deux types de sommets : les **places** et les **transitions** reliées alternativement par des **arcs orientés** qui portent des poids entier positifs, si un poids n'est pas porté alors il est égal à **1** (**RdP ordinaire**). Généralement, les places sont représentées par des cercles et les transitions par des rectangles, le marquage d'un **RdP** est représenté par la distribution de jetons dans l'ensemble de ses places telle que chaque place peut contenir un ou plusieurs jetons représentés par des points dans le cercle représentant la place[35].

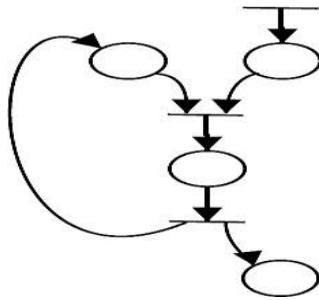


Figure IV-1 Réseau de Petri simple[35].

IV.4.1.2. Représentation matricielle

Une représentation matricielle d'un RdP est offerte afin de simplifier les tâches d'analyse et de vérification effectuée sur un modèle RdP. Agir sur une représentation graphique d'un modèle RdP est une tâche délicate en comparant avec une représentation matricielle.

Il est possible de représenter la fonction W (fonction de poids) par des matrices.

Définition

Soit Un réseau de Pétri $R = (P, T, W)$, on appelle matrice des pré conditions $pré$ la matrice $m \times n$ à coefficients dans \mathbb{N} tel que $pré(i, j) = W(p_i, t_j)$, elle indique le nombre de marque que doit contenir la place P_i pour que la transition t_j devienne franchissable, de la même manière on définit la matrice des post :

$post$ la matrice $n \times m$ tel que $post(i, j) = W(t_j, p_i)$ contient le nombre de marques déposées dans P_i lors du franchissement de la transition t_j . La matrice $C = post - pré$ est appelée matrice d'incidence du réseau (m représente le nombre de places d'un **réseau de Pétri** et n le nombre de transitions[36]).

Le marquage d'un réseau de Pétri est représenté par un vecteur de dimension m à coefficients dans N . La règle de franchissement d'un réseau de Pétri est définie par

$M'(p) = M(p) + C(p, t).$	Équation IV-2
---------------------------	---------------

IV.4.1.3. Représentation d'un RdP marqué

Un réseau de Pétri marqué est le couple $N = \langle R, M \rangle$ où :

- R est un réseau de Pétri.
- M est une application de marquage $M : P \rightarrow N$.

$M(p)$ est le nombre de marques (jetons) contenus dans la place p . Le marquage d'un réseau de Pétri est une opération qui consiste à assigner des jetons dans les places.

On appelle marquage M d'un Réseau de Pétri le vecteur du nombre de marques dans chaque place : la $i^{\text{ème}}$ composante correspond au nombre de marques dans la $i^{\text{ème}}$ place[37].

Exemple

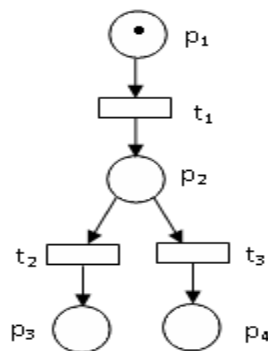


Figure IV-2 Réseau de Petri marqué[37].

Pour le réseau de la ($P = \{p1, p2, p3, p4\}$ $T = \{t1, t2, t3\}$)

$$Pré = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$post = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

La matrice d'incidence est :

Le vecteur de marquage M est :

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

$$M = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

On appelle marquage initial, noté M_0 , le marquage à l'instant initial ($t = 0$).

IV.5. L'aspect comportemental

Le comportement d'un réseau de Pétri est déterminé par sa structure et par son état. Pour exprimer l'état d'un réseau de Pétri, les places peuvent contenir des jetons qui ne sont que de simples marqueurs[37].

IV.5.1. L'état dans un réseau de Pétri

Dans la théorie des réseaux de Pétri, l'état d'un réseau est souvent appelé *marquage* du réseau qui est défini par la distribution des jetons sur les places. Le comportement d'un **réseau de Pétri** marqué est déterminé par ce qu'on appelle **règle de franchissement**[38].

IV.5.2. Franchissement d'une transition

Une règle de franchissement est une simple relation de transition qui définit le changement d'état dans un réseau marqué lors de l'exécution d'une action. Afin de définir une règle de franchissement, il est nécessaire de formaliser quand le réseau peut exécuter une action : on dit qu'une transition peut être **franchie** à partir d'un marquage M (qui représente l'état du system à un instant donné) si et seulement si chaque place p d'entrée de la transition t contient au moins un nombre de jetons qui est supérieur ou égal au poids de l'arc reliant cette place d'entrée p avec la transition t tel que :

$M(p) \geq W(p, t) \forall p \in P$	Équation IV-3
-------------------------------------	---------------

Une règle de franchissement est définie pour tout par :

$M'(p) = M(p) - W(p, t) + W(t, p)$	Équation IV-4
------------------------------------	---------------

ce qui veut dire que lorsque la transition t est franchi à partir d'un marquage M , il faut saisir $W(p, t)$ jetons à partir de chaque place d'entrée à la transition t et déposer $W(t, p)$ jetons dans chaque place de sortie de la transition t ce qui permet de produire un nouveau marquage M' .

Le franchissement d'une transition t dénoté par $M \rightarrow M'$ est dite l'**occurrence** de t . On dit que deux transitions t_1, t_2 (pas certainement distinctes) sont franchies en concurrence par un marquage M si et seulement si $M(p) \geq W(p, t_1) + W(p, t_2)$ pour toute $p \in P$.

Cette vision de l'exécution concurrente de deux transitions dans un RdP est contradictoire avec celle qui impose que deux occurrence de transition sont parallèles si et seulement si : elles sont causalement indépendantes et n'ont pas une relation de conflit entre eux. Deux occurrences sont en conflit si l'un des deux peut avoir lieu mais pas toutes les deux[39].

IV.5.3. L'exécution d'un réseau de Petri

IV.5.3.1. Exécution séquentielle

- **Séquence de franchissement**

Une séquence de franchissement « s » est une suite de transitions (t_1, t_2, \dots, t_n) qui permet, à partir d'un marquage « M », de passer au marquage « M' » par le franchissement successif des transitions définissant la séquence.

- **Marquage accessible**

Le marquage d'un Réseau de Petri à un instant donné est une vectrice colonne dont la valeur de la $i^{\text{ème}}$ composante est le nombre de marques de la place P_i à cet instant.

Le franchissement d'une transition conduit à un changement du marquage. Le passage du marquage M_k au marquage M_l par franchissement de la transition T_j est noté : $M_k \xrightarrow{T_j} M_l$. Le nombre de marques dans la place P_i pour le marquage M_k est noté $M_k(P_i)$. A partir d'un même marquage, il peut être possible de franchir plusieurs transitions, menant ainsi à des marquages différents. L'ensemble des marquages accessibles à partir du marquage M_0 est l'ensemble des marquages obtenus à partir de M_0 par franchissements successifs d'une ou plusieurs transition(s). Cet ensemble est noté $A(R, M_0)$ [40].

- **Graphe de marquage**

On peut représenter l'ensemble de marquage accessible par un graphe si ce dernier est fini. Le graphe de marquage a comme sommet l'ensemble de marquage accessible $A(R, M_0)$. Un arc orienté relie deux sommets M_i et M_j s'il existe une transition t franchissable permettant de passer d'un marquage à un autre $M_i \xrightarrow{t} M_j$. Les arcs du graphe sont étiquetés par les transitions correspondantes.

Exemple :

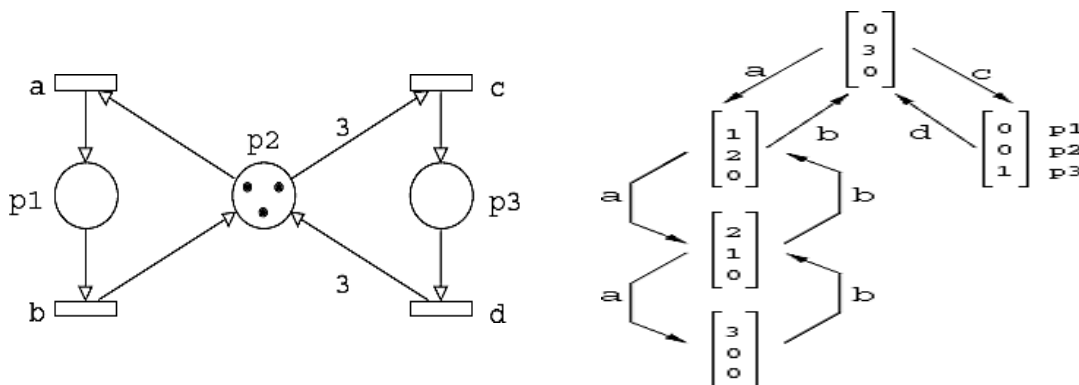


Figure IV-3 Graphe de Marquage

- **L'exécution séquentielle d'un réseau de Petri**

L'exécution séquentielle d'un réseau de Petri est définie en termes d'un ensemble de séquences d'occurrence. Une séquence d'occurrence est une séquence de transitions franchissables dénotée par $\sigma = M_0 t_1 M_1 t_2 \dots$ tel que $M_{i-1} [t_i > M_i$. Une séquence $t_1 t_2 \dots$ est une séquence de transitions (commencée par le marquage M) si et seulement s'il existe une séquence d'occurrence $M_0 t_1 M_1 \dots$ avec $M = M_0$. Si la séquence finie $t_1 t_2 \dots t_n$ conduit à un nouveau marquage M' à partir du marquage M , on écrit $M [t_1 t_2 \dots t_n > M'$ ou simplement $M [t_1 t_2 \dots t_n >$ si on ne veut pas spécifier le marquage résultat [41].

- **Exécution concurrente**

Une exécution concurrente d'un réseau de Pétri est une exécution dans laquelle plusieurs transitions peuvent se franchir au même temps, elle est souvent déterminée par la notion de processus. Ceci permet de donner une interprétation de la concurrence dans un réseau de Pétri selon la sémantique basée sur la vraie concurrence (sémantique d'ordre partiel) qui est interprétée dans la théorie des réseaux de Pétri par un type spécial de réseaux appelés réseaux d'occurrences.

IV.6. Réseaux particuliers

Le graphe associé à un réseau de Pétri peut être très complexe. Un certain nombre de situations présentent un intérêt particulier :

IV.6.1. Graphe d'états

Dans ce cas chaque transition ne dispose que d'une place en entrée et une place en sortie.

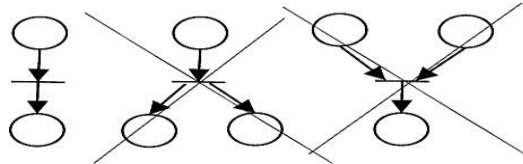


Figure IV-4 Graphe d'états ou pas.

IV.6.2. Les réseaux sans conflits

Un réseau de Petri est dit sans conflit si et seulement si toute place a *au plus une* transition de sortie (voir Figure IV-5). Un conflit (structurel) correspond à l'existence d'une place P_i qui a au moins deux transitions de sortie T_j, T_k, \dots . Notation $\langle P_i, \{T_j, T_k, \dots\} \rangle$. Sur le RdP de droite de la (Figure IV-5), on a le conflit $P_1, \{T_2, T_3\} \rangle$. Quand la place P_1 contient une marque, les transitions T_2 et T_3 sont franchissables. Seule une des deux transitions peut être franchie : il est nécessaire de prendre une décision pour savoir laquelle des deux le sera effectivement. L'absence ou la présence d'un conflit est une propriété importante d'un réseau de Petri.

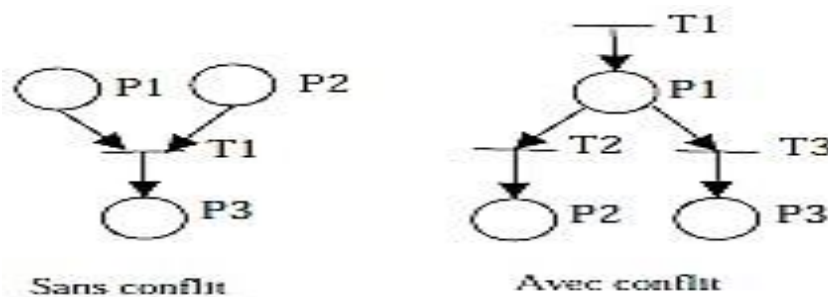


Figure IV-5 Conflit ou pas.

IV.6.3. Les réseaux purs

Un réseau de Pétri est dit pur si et seulement si il n'existe pas de transition ayant une place d'entrée qui est aussi place de sortie. Le RdP représenté (Figure IV-6) n'est pas pur car la place P_3 est place d'entrée et place de sortie de la transition T_1 . On parle alors de RdP impur.

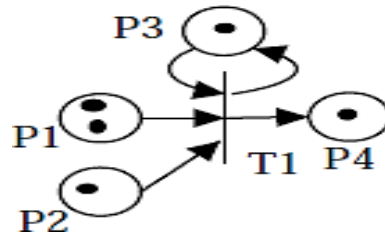


Figure IV-6 RdP impur.

IV.7. Propriétés des RdP

IV.7.1. Réseau K-borné

Une place P_i est bornée pour un marquage initial M_0 si pour tout marquage accessible à partir de M_0 , le nombre de marques dans P_i reste borné. Elle est dite **k-bornée** si le nombre de marques dans P_i est toujours inférieur ou égal à k . Un RdP marqué est (k) borné si toutes ses places sont (k) bornées.

Un RdP marqué peut ne pas être borné : sur l'exemple représenté, la transition T_1 admet la place P_1 comme unique place d'entrée. La place P_1 a une marque : la transition T_1 est franchissable. Comme P_1 est aussi place de sortie de T_1 , le franchissement de T_1 ne change pas le marquage de P_1 . La transition T_1 est donc franchissable en permanence et peut donc être franchie un nombre de fois infini. Chaque franchissement de T_1 ajoutant une marque dans la place P_2 , le marquage de celle-ci peut donc tendre vers l'infini.

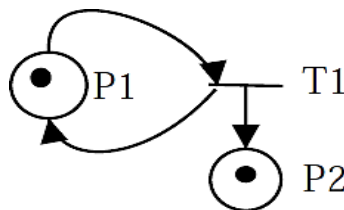


Figure IV-7 RdP non borné.

Définition :

Un RdP marqué est sauf ou binaire pour un marquage initial M_0 s'il est 1-borné.

IV.7.2. Réseau vivant

L'évolution du marquage d'un RdP se fait par franchissement de transitions. Lorsqu'au cours de son évolution, certaines transitions ne sont jamais franchies, cela indique que l'évènement associé à la transition ne se produit pas et que le marquage d'une partie du RdP n'évolue pas. Cela indique que le sous-système modélisé par cette partie-là ne fonctionnera pas. Il y a donc un problème au niveau de la conception du système. L'idée est d'être capable de détecter systématiquement ce phénomène par l'analyse de propriétés du modèle RdP du système afin de disposer d'un outil d'aide à la conception des systèmes.

Définition : Une transition T_j est vivante pour un marquage initial M_0 si pour tout marquage accessible M_k , il existe une séquence de franchissements à partir de M_k contenant T_j :

$$\forall M_k \in {}^*M_0, \exists S, M_k \mid S > \text{ et } S = \dots T_j \dots$$

Équation IV-5

Si une transition T_j est vivante alors, à tout instant, on sait que T_j peut être franchie dans le futur. Dans le cas d'un RdP modélisant un système fonctionnant en permanence, si une transition n'est pas vivante et si une fonction du système est associée au franchissement de cette transition, cela veut dire qu'à partir d'un certain instant, cette fonction ne sera plus disponible dans le futur, ce qui peut traduire une erreur ou une panne.

Exemples : Les transitions T_1 et T_2 du RdP marqué (Figure IV-8) sont vivantes

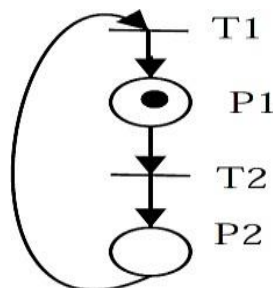


Figure IV-8 RdP vivant.

IV.7.3. Réseau réinitialisable

Soit un réseau de Petri marqué $\langle R; M_0 \rangle$ et soit $A(R, M_0)$ l'ensemble de ses marquages accessibles. Ce réseau marqué est réinitialisable si et seulement si :

$$\forall M \in A(R; M_0), M \neq M_0, \exists s \text{ tel que } M \xrightarrow{s} M_0$$

Équation IV-6

On dit également que M_0 est un état d'accueil (en anglais "home state") pour le réseau de Petri marqué.

Soit un réseau de Petri marqué $\langle R; M_0 \rangle$, s'il est réinitialisable et que toutes ses transitions sont **quasi-vivantes** ou **vivantes**, alors il est vivant

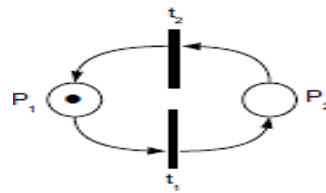


Figure IV-9 Réseau réinitialisable.

IV.8. Les méthodes d'analyse des RdP

Les méthodes d'analyse des RdP peuvent être classées en trois groupes :

- Méthode d'arbre de couverture
- Approche d'équations matricielles
- Technique de réduction et de décomposition

IV.8.1. Méthode d'arbre de couverture

L'idée la plus naturelle pour étudier les propriétés d'un RdP est de construire le graphe de tous les marquages accessibles. Le graphe des marquages accessibles est un graphe dont chaque sommet correspond à un marquage accessible et dont chaque arc correspond au franchissement d'une transition permettant de passer d'un marquage à l'autre.

Pour un RdP marqué (R, M_0) , à partir d'un marquage initial M_0 on peut obtenir autant de nouveaux marquages qu'il existe de transitions franchissables. À partir de chaque nouveau marquage on peut accéder à d'autres nouveaux marquages.

Le résultat de ce processus est un arbre dont chaque nœud est un marquage accessible et chaque arc est une transition franchissable qui transforme un marquage à un autre

L'arbre de couverture est une méthode alternative. Le mécanisme de construction est le même que pour le graphe des marquages accessibles. A ceci près que pour chaque nouveau marquage (nœud du graphe) ajouté, on vérifie s'il n'est pas supérieur à un marquage déjà présent sur au moins une séquence entre M_0 et le nouveau marquage. Si tel est le cas tous les marquages de place supérieurs sont remplacés par ω . Ce symbole matérialise le fait que la place en question peut contenir autant de jetons que souhaité (elle est donc non bornée).

Dans la suite, les places non bornées le demeurent naturellement et ceci quelles que soient les transitions franchies, ainsi le symbole ω ne disparaît jamais. Cette méthode produit le graphe de couverture, un graphe fini dans tous les cas. [42].

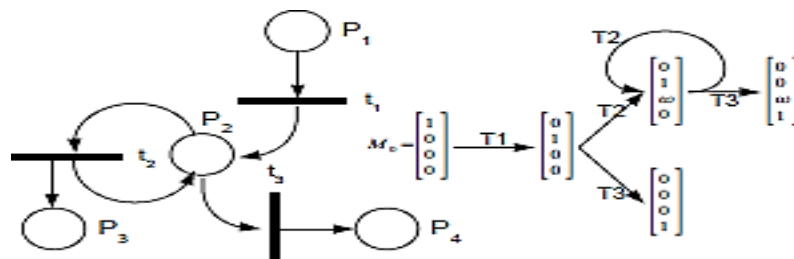


Figure IV-10 RDP et leur d'arbre de couverture

IV.8.2. Approche d'équations matricielles

L'analyse par équations matriciel (algèbre linéaire) permet d'étudier des propriétés d'un réseau (caractère borné, vivacité) indépendamment d'un marquage initial. De ce fait, on parlera de propriétés structurelles du réseau. Par exemple, on pourra dire qu'un réseau est structurellement borné s'il est borné pour tout marquage initial fini. De la même façon, si pour tout marquage initial, le réseau est vivant, on dira que le réseau est structurellement vivant.

IV.8.3. Technique de réduction et de décomposition

L'analyse des RDPs par réduction permet d'obtenir à partir d'un RdP marqué, un RdP marqué plus simple, c'est-à-dire avec un nombre réduit de places et un nombre réduit de transitions ceci en appliquant un ensemble de règles dites règles de réduction.

IV.9. Réseaux de Pétri stochastiques généralisés (RdPSG)

Le modèle des RDPSG est apparu afin de résoudre les problèmes liés à la présence de divers types d'événements dans un même modèle, tels que les activités qui ne nécessitent pas de temporisation et d'autres qui sont très rapides (urgentes) ou très lentes. Le modèle des RDPSG prend en considération la nature de ces activités, afin que le modèle soit logiquement correct.

Ainsi, ce modèle est caractérisé par l'introduction d'un nouveau type de transitions, appelées transitions temporisées. Par conséquent, un RDPSG est un RdP dont les transitions sont de deux types :

- **Transitions temporisées** : à qui correspondent les variables aléatoires déterminant la durée de franchissement.
- **Transitions immédiates (instantanées)** : qui se caractérisent par une période de franchissement supérieure à celle des transitions temporisées, et par leur franchissement immédiat, car le taux de franchissement associé est infini, ainsi le délai de tir correspondant à ces transitions est nul.

Définition[43] :

Un RDPSG est un triplet $\langle R_0, \pi, W \rangle$ où :

- R_0 Est un RdP simple.
- π Est la fonction de priorité associant un entier à toute transition.
- W Est une fonction qui associe à chaque transition temporisée un taux de franchissement et à chaque transition immédiate un poids.

IV.9.1. Evolution d'un RDPSG

Les marquages accessibles dans un RDPS se décomposent en deux catégories :

- **Marquages tangibles** : Dans lesquels, aucune transition immédiate n'est franchissable.

Si tous les marquages sont tangibles, alors le processus stochastique engendré par un RDPSG est identique au RDPS.

- **Marquages évanescents** : Dans lesquels, il y a au moins une transition immédiate parmi les transitions franchissables. Dans ce cas, la priorité est donnée aux transitions immédiates car leur franchissement se fait en un temps nul. D'autre part, si un marquage évanescents sensibilise plusieurs transitions immédiates à la fois, le choix de celle qui sera franchissable ne peut se baser sur des considérations temporelles, puisque la durée de leur franchissement est nulle, mais il se fait par

une post-sélection. Cette post-sélection, de nature probabiliste, est basée sur les poids des transitions donnée par l'expression suivante :

$P\{t_k\} = W_k / \sum W_i$	Équation IV-7
-----------------------------	---------------



Talque :

W_k est le poids de la transition t_k et la somme est faite sur l'ensemble des transitions immédiates sensibilisées par le marquage M .

Le marquage obtenu par le franchissement d'une transition t , est calculé de la même manière qu'un RdPs:

$M'(\cdot, t) = M(\cdot, t) + C(\cdot, t); \text{ tel que: } C(\cdot, t) = -\text{Pré}(\cdot, t) + \text{Post}(\cdot, t);$	Équation IV-8
--	---------------

IV.9.2. Représentation graphique d'un RDPSG

Les RDPSG sont constitués de transitions temporisées stochastiques représentées par des rectangles  et de transitions *immédiates* représentées par des barres noires  fines :

IV.9.3. Propriétés des RDPSG [44]

Nous pourrions examiner le RDP sous-jacent pour déterminer les propriétés du RDPSG. Autrement dit, toutes les transitions qu'elles soient temporisées ou immédiates, sont traitées de la même façon lors de l'analyse qualitative.

IV.9.4. Analyse quantitative

L'intérêt des RdPS réside particulièrement dans la nature du graphe des marquages qui est équivalent à une chaîne de Markov homogène. De ce fait, les méthodes de calcul des processus Markoviens deviennent applicables au calcul des paramètres de performances. On dit alors que le graphe des marquages d'un RDPS est isomorphe à une chaîne de Markov homogène.

L'analyse quantitative consiste à calculer les paramètres de performances. Ceci est possible grâce à l'isomorphisme entre les RDPS et les chaînes de Markov.

Nous pouvons donc évaluer les performances du système modélisé tout en appliquant les étapes suivantes [45] :

- Construire d'abord la matrice des taux de transitions.
- Ensuite résoudre le système matriciel afin de calculer les probabilités d'états en régime permanente.

IV.9.4.1. Condition d'ergodicité

L'analyse quantitative à l'état d'équilibre nécessite la vérification de la condition d'ergodicité.

Les modèles ergodiques sont les plus intéressants car cette propriété assure l'existence d'un régime stationnaire du système modélisé permettant le calcul d'une solution stationnaire [46, 47].

Théorème

Un RDPS borné est ergodique, s'il admet l'état initial comme état d'accueil [48] .

IV.9.4.2. Evaluation des performances d'un RDPS

Une fois l'ergodicité du modèle soit vérifiée, nous pouvons alors évaluer les performances du système modélisé en procédant de la manière suivante [49]:

- Construire la matrice des taux de transition Q, dite *générateur infinitésimal*.
- Calculer le vecteur des probabilités d'états stationnaires.
- Calculer des paramètres quantitatifs à l'aide du vecteur des probabilités stationnaires qui sont les suivants :
 - Fréquence moyenne de franchissement d'une transition.
 - Nombre moyen de marques dans une place.
 - Temps moyen de séjour des marques dans une place.
 - Probabilité d'un événement A défini à travers une condition.

Ces quantités peuvent représenter respectivement dans le cas d'un exemple de RDPS modélisant un protocole de communication, le débit du canal de transmission, la charge du canal et le délai moyen de transmission.

IV.9.4.3. Calcul du générateur infinitésimal [46]

Les éléments de la matrice Q sont calculés à l'aide du graphe des marquages accessibles. Ils correspondent aux taux de transition entre états ou marquages. La matrice est carrée, d'ordre égal au nombre de marquages dans le graphe d'accessibilité. Les taux de transition sont définis comme suit :

$Q[i, j] = \begin{cases} \lambda_{ij} & \text{si } i \neq j \\ -\sum_{\substack{k=1 \\ k \neq i}}^n Q[i, k] & \text{si } i = j \end{cases}$	Équation IV-9
---	---------------

Où : n correspond au nombre de marquages, et λ_{ij} désigne le taux de la transition du marquage M_i au marquage M_j .

Les autres éléments de la matrice sont nuls.

IV.9.4.4. Calcul du vecteur des probabilités stationnaires [46]

Le vecteur π des probabilités stationnaires (en régime permanent), de dimension égale au nombre de marquages du système modélisé, est calculé en résolvant le système d'équations suivant :

$\begin{cases} \pi Q = 0 \\ \sum_{i=1}^n \pi_i = 1 \end{cases}$	Équation IV-10
---	----------------

IV.9.5. Critères de performances [46, 47]

Le vecteur π des probabilités stationnaires décrit ci-dessus va nous aider dans le calcul des paramètres de performances comme suit :

IV.9.5.1. Fréquence moyenne de franchissement d'une transition t_i

$N^*(t_i) = \sum_{M_j \in GMA} \lambda_i(M_j) \cdot \pi_j$	Équation IV-11
--	----------------

Où : $\lambda_i(M_j)$ est le taux de franchissement de la transition t_i dans le marquage M_j .

IV.9.5.2. Nombre moyen de marques dans une place p

$N^*(p) = \sum_{M_j \in GMA} M_j(p) \cdot \pi_j$	Équation IV-12
--	----------------

Où : $M_j(p)$ est le nombre de marques de la place p pour le marquage M_j .

IV.9.5.3. Temps moyen de séjour des marques dans une place p

$T^*(p) = M^*(p) / (Post(p, \cdot) * N^*)$	Équation IV-13
--	----------------

Où : N^* est le vecteur des fréquences moyennes de franchissement des transitions.

$Post(p, .)$ est la ligne de la matrice d'incidence arrière $Post$ correspondant à la place p .

$M^*(p)$ est le nombre moyen de marques dans la place p .

IV.9.5.4. Probabilité d'un événement A défini à travers une condition

$Prob(A) = \sum \pi_i$	Équation IV-14
------------------------	----------------

La somme est effectuée sur les indices des marquages où la condition est satisfaite.

Remarque

Il existe certains paramètres de performances qui sont dans le cas général difficile à calculer comme le calcul des distributions des délais de franchissement [43].

IV.10. Outils de modélisation des RDPs

L'aspect formel des RDPs a encouragé les développeurs à mettre au point une multitude d'outils de simulation et de vérification des RDPs selon la technique de vérification de modèle. On propose de faire un résumé sur les fonctionnalités qu'offre chacun d'eux Le tableau ci-dessous les classe par rapport aux points suivants[50]:

- ✓ Présence d'un environnement graphique d'édition des RDPs.
- ✓ Possibilité de Simulation du RDP.
- ✓ Possibilité d'analyse des propriétés génériques du RDP.
- ✓ Possibilité de vérification des contraintes CTL (logique temporelle arborescente)
- ✓ Et LTL (logique temporelle linéaire).
- ✓ Possibilité de supporter le format d'échange XML.
- ✓ Possibilité de supporter les RDP hiérarchiques.

Outil	InterfaceGraphique	Simulation	Analyse	CTL	LTL	XML	RDPHiérarchique	TOTAL(oui)
CPNTools	Oui	Oui	Oui	Oui	Non	Oui	oui	6
CPNAMI	Oui	Oui	Oui	Oui	Oui	Non	non	5
PROD	Non	Non	Oui	Oui	Oui	Non	non	3
JARP	Oui	Non	Oui	Non	Non	Oui	non	3
Maria	Non	Non	Oui	Oui	Oui	Non	non	3
LoLa	Non	Non	Oui	Oui	Non	Non	non	2
Petri NetKernel	Oui	Non	Non	Non	Non	Oui	non	2
Great SPN	Oui	Non	Oui	Non	Non	Non	Non	2
INA	Non	Oui	Oui	Oui	Non	Non	Non	2
OPMSE	Oui	Oui	Non	Non	Non	Non	non	2
TimeNet	Oui	oui	oui	-	-	oui	-	4

Tableau IV-1 Les outils de modélisation des réseaux de Petri[50].

IV.11. Conclusion

Dans ce chapitre nous avons donné les définitions de base d'un réseau de Petri. Bien que simples, ces définitions se compliquent lorsque l'on va au bout des choses.

Dans ce qui suit, nous utiliserons l'outil TimeNet pour étudier le RdP proposé pour les WSNs

Chapitre 4

Modélisation et Analyse d'un WSN avec les Réseaux de Pétri

V.1. Introduction

Un nœud de capteur (SN) est un petit appareil capable de collecter des données de l'environnement et de les envoyer via un réseau de communication sans fil.

Le principal problème de ces capteurs est leur courte durée de vie, déclenchée par la capacité limitée de leurs batteries. Une technique relativement nouvelle consiste à utiliser des piles rechargeables pour récupérer l'énergie de l'environnement (soleil, vent, chaleur, pression, etc.) et à la convertir en énergie électrique.

Ce type de réseau s'appelle Réseaux de Capteur sans fil avec récupération d'énergie[51].

Afin d'être plus précis, l'évaluation de la performance de ce type de réseau doit prendre en compte plusieurs circonstances importantes pour décrire de manière fiable le comportement réel du réseau. Ces circonstances sont les suivantes : rappels, mécanisme de veille également appelé serveur avec vacances, récupération d'énergie, pertes de messages, etc.

Il existe deux manières d'évaluer les performances d'un réseau : la simulation et la modélisation. Pour la modélisation, nous utilisons des approches formelles et mathématiques pour obtenir des résultats de performance.

Dans ce chapitre, nous utilisons le formalisme RDPSG (GSPN) pour traiter le phénomène de rappel, mécanisme de veille et récupération d'énergie. Ce modèle permet d'ajuster les paramètres d'entrée afin de déterminer les valeurs qui assurent la vivacité et le bon comportement du réseau.

V.2. Un modèle GSPN pour WSNs

En considérant la relation Capteur-voisins, un message arrivé au capteur principal (Capteur modélisé) prend le service directement s'il y a un voisin libre, il est enregistré dans le buffer sinon. Si le buffer est plein, le message rejoint l'orbite. Le franchissement de la transition *Envoyer_msg* consomme un quantum d'énergie. La transition *Harvesting* représente la fonction de récupération d'énergie. La figure V-1 représente notre modèle proposé.

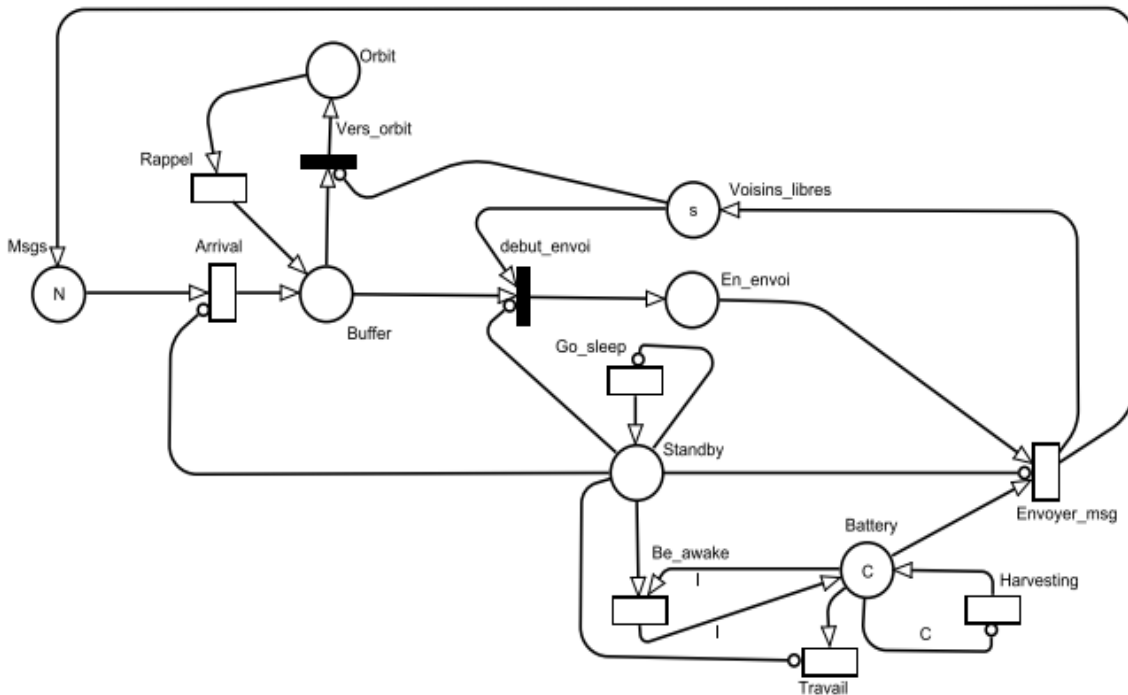


Figure V-1 Modéle GSPN du réseau de capteur sans fil

Cependant le SN principal peut être actif ou en veille. Lorsque le SN est en attente (C'est-à-dire qu'un jeton est à la place *Standby*), aucune opération ne peut être effectuée dans le SN (voir les arcs inhibiteurs). Le modèle proposé contient deux **transitions immédiates** : *début_envoi* représente le début de l'opération de transmission après le choix d'un voisin libre et le déclenchement de la transition *Vers_orbit* rejoint le message dans l'orbite lorsque la mémoire tampon est pleine. Les **transitions temporisées** sont décrites dans le tableau V-1. Le tableau V-2 décrit les places du modèle.

Transition	Type	Signification	Taux de franchissement
<i>Arrival</i>	Temporisée	Arrivée d'un message	λ
<i>Envoyer_msg</i>	Temporisée	Envoi réussi d'un message	μ
<i>Rappel</i>	Temporisée	Nouveau rappel d'un message	ν
<i>Harvesting</i>	Temporisée	Récupération d'énergie	ω
<i>Travail</i>	Temporisée	Consommation d'énergie dans l'état actif	γ
<i>Go_sleep</i>	Temporisée	Le capteur principal rejoint l'état de veille	α
<i>Be_awake</i>	Temporisée	Activation du capteur principal	δ
<i>Debut_envoi</i>	Immédiate	commencer à envoyer le message	θ
<i>Vers_orbit</i>	Immédiate	transférer les message vers l'orbite	ε

Tableau V-1 Description des transitions temporisées

Nom de Place	La description	Valeur initiale
<i>Msgs</i>	Source d'un message	N
<i>buffer</i>	Mémoire tampon	0
<i>Orbite</i>	Un message dans cette place rappelle à obtenir une réception	0
<i>Voisins_libres</i>	Contient les voisins libre	s
<i>En_envoi</i>	Contient les messages à envoyer	0
<i>Battery</i>	Un jeton dans la batterie représente un quantum d'énergie	C
<i>Standby</i>	SN principal est en état de veille	0

Tableau V-2 Description des Places

V.1. Formules de Performance

A l'état stationnaire, plusieurs formules de performance peuvent être définies à l'aide de formalisme GSPN. La charge moyenne de la batterie noté par $\overline{Battery}$. Il correspond au nombre moyenne de jetons dans la place *Batterie* :

$\overline{Battery} = \sum_{i: M_i \in M} M_i (Battery). \pi_i$	Équation V-1
---	--------------

Tel que π représente le vecteur de probabilité stationnaire obtenu après la résolution du système.

L'application de la loi de *Littles* prouvée dans [52] permet de calculer le temps d'attente d'un jeton dans une place. Par exemple, le temps d'attente d'un jeton dans la place *Orbit* indiqué par :

- **Wait (Orbit)** est égal au nombre moyen de jetons dans cette place divisée par \bar{v} :

\bar{v} Est le débit de la transition **Rappel**. Donc :

$Wait (Orbit) = \frac{\overline{Orbit}}{\bar{v}}$	Équation V-2
---	--------------

Tel que :

- \overline{Orbit} Est le nombre moyen de marques dans la place *Orbit*:

$\overline{Orbit} = \sum_{i: M_i \in M} M_i (Orbit). \pi_i$	Équation V-3
---	--------------

- **Wait (Enenvoi)** est égal au nombre moyen de jetons dans cette place divisée par $\bar{\mu}$:

$$Wait(En_envoi) = \frac{\overline{En_envoi}}{\bar{\mu}}$$

Équation V-4

Tel que :

- $\overline{En_envoi}$ Est le nombre moyen de jeton dans la place *En_envoi*, donné par la formule suivante :

$$\overline{En_envoi} = \sum_{i:M_i \in M} M_i(En_envoi) \cdot \pi_i$$

Équation V-5

- $\bar{\mu}$ représente le débit de la transition *En_envoi*, donné par la formule suivante :

$$\bar{\mu} = \frac{1}{\mu} \sum_{i:M_i \in M} M_i(En_envoi) \cdot \pi_i$$

Équation V-6

- \overline{Buffer} Est le nombre moyen de jeton dans la place *buffer*, donné par la formule suivante :

$$\overline{Buffer} = \sum_{i:M_i \in M} M_i(buffer) \cdot \pi_i$$

Équation V-7

- Le temps de réponse moyen pour envoyer un message *R* est donné comme suit:

$$R = \frac{\overline{En_envoi} + \overline{Buffer} + \overline{Orbit}}{\bar{\mu}}$$

Équation V-8

Tel que :

$\bar{\mu}$ est le débit de la transition *En_envoi*. Sur la base du support mathématique du GSPN, d'autres paramètres de performance peuvent être définis [53].

V.2. Conclusion

Dans ce chapitre vous avez présenté notre modèle GSPN pour les réseaux de capteurs sans fils à l'aide des réseaux de Petri, on a bien modélisé et présenté la structure de circulation d'information (un message) entre les différents capteurs avec une technique de consommation d'énergie *Mode de veille* pour minimiser la consommation d'énergie et la récupération d'énergie pour recharger les batteries des capteurs.

On a terminé avec une petite analyse de performances sur notre modèle basé sur GSPN en état stationnaire, utilisant des équations mathématiques pour évaluer les performances de notre réseau.

Et dans le chapitre suivant en montrant les résultats d'évaluation et la simulation sur notre GSPN.

Chapitre 5

Application et étude expérimentale

VI.1. Introduction

Dans le cadre de cette étude expérimentale, les résultats numériques ont été calculés en utilisant l'outil software *TimeNet*, qui est un outil très performant, dédié à l'évaluation des performances. *TimeNet* offre à l'utilisateur un environnement graphique simple lui permettant de définir tous les éléments du modèle, ainsi que tous les paramètres de performance et de fiabilité désirés.

Ainsi, en se basant sur cet outil performant, nous obtenons des résultats qui sont présentés sous forme de graphes qui sont commentés et à partir desquels des conclusions seront tirées.

VI.2. Description du logiciel *TimeNet*

TimeNet est un progiciel et environnement graphique interactif qui permet de modéliser différents systèmes avec les méta-modèles de réseaux de Pétri stochastiques généralisés et les réseaux de Pétri stochastiques colorés (SCPNS). Dans ce qui suit, nous allons présenter des fonctionnalités essentielles en se basant sur la documentation fournie avec le logiciel *TimeNet* [54].

VI.2.1. Historique du logiciel *TimeNet*

TimeNET a été développé par l'équipe de *systèmes temps réel* et *groupe de Robotics* de l'université de Berlin en Allemagne. Le projet a été motivé par le besoin d'un logiciel puissant pour l'évaluation efficace des réseaux de Pétri stochastiques.

La première version du *TimeNet* était une révision majeure de l'outil *DSPN express* à l'université de Berlin. Elle contenait toutes les composantes d'analyses existantes maintenant mais pour la classe eDSPNs (extended Deterministic Stochastic Petri Nets : RDPS déterministes) seulement.

VI.2.2. Fonctionnalités du *TimeNet*

Différentes solutions algorithmiques peuvent être utilisées en fonction de la classe de RdPs. La nouvelle version du *TimeNet*. Il comporte une interface graphique spéciale pour la classe des RdPSC. Cette interface fournit différentes fonctionnalités liées à cette classe du réseau de Pétri. ***TimeNet*** peut calculer la solution du modèle dans le régime stationnaire et fournit une technique d'analyse approximative pour la classe eDSPNs qui comporte les RdPSG et les RdPSD.

L'analyse, l'approximation et la simulation peuvent être effectués pour une même classe du modèle. En plus les enrichissements du Time Net permis lesquels nous citons :

- ✓ Un algorithme permettant l'analyse transitoire pour la classe RdPSD.
- ✓ Un composant pour le régime stationnaire et l'analyse transitoire pour les RdPSG.
- ✓ Un composant spécial désigné aux systèmes de fabrication.
- ✓ Une interface graphique d'utilisateur complet qui intègre différentes classes de RdP et permet d'utiliser facilement différents algorithmes d'analyse.

VI.2.1. Interface graphique du *TimeNet*

La nouvelle version du *TimeNet* comporte une interface graphique interactive développée à l'aide du langage *JAVA* et fournit un support du système d'exploitation *Windows*. Le *TimeNet* supporte également maintenant la classe des RDPSC (RDPS Colorés). L'interface graphique générique du *TimeNet* permet la visualisation des résultats de simulation et de l'analyse sous forme de graphes.

La fenêtre de l'interface graphique du *TimeNet* est divisée en quatre parties comme le montre la Figure VI-1 :

- ✓ Une barre de menu en haut.
- ✓ Une zone de dessin à gauche.
- ✓ Une zone des attributs à droite.
- ✓ Une barre d'outils spécifique à la classe du RDP en bas.

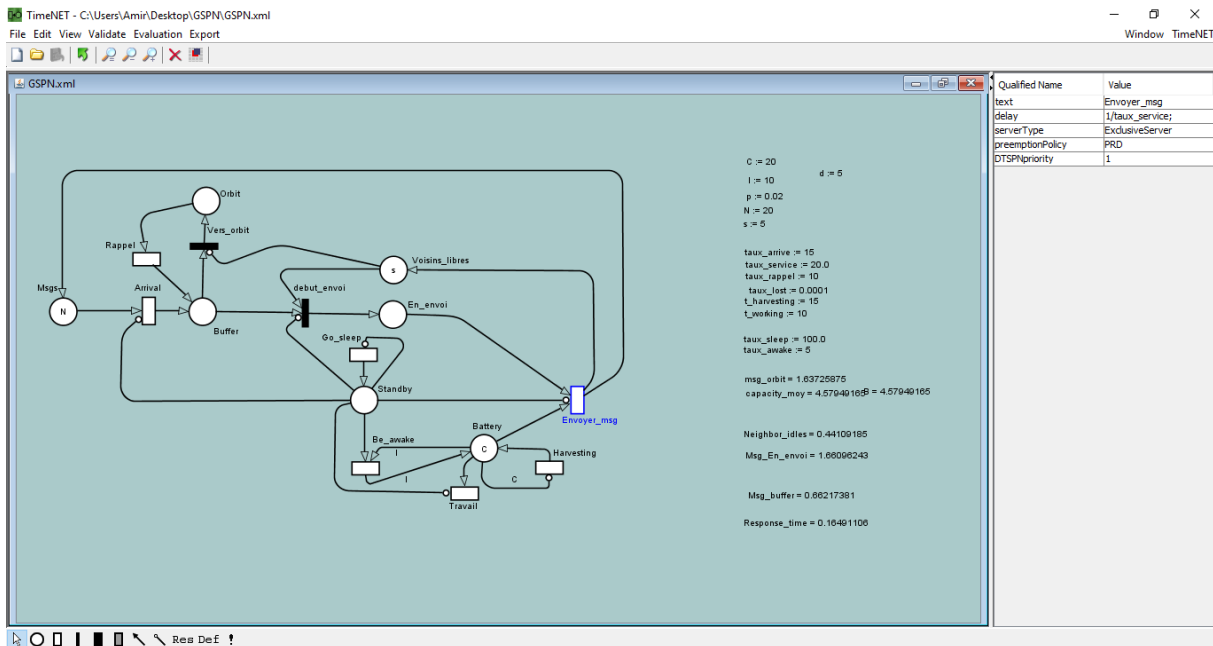


Figure VI-1 L'interface graphique du TimeNET sous Windows

VI.3. Mesure de performances

Pour mesurer un paramètre de performance, il suffit de placer un objet « $R =$ » à la surface du dessin. Une mesure de performance définit ce qui va être calculé durant l'analyse. La valeur d'une mesure peut être par exemple le nombre moyen de jetons dans une place.

Une mesure a les attributs suivants :

- ✓ Le nom du paramètre mesuré.
- ✓ Une expression qui doit être évalué.
- ✓ Le résultat qui remplace l'expression après une évaluation réussite du modèle.

Une mesure est une expression qui peut contenir des nombres, des taux de franchissement, des opérateurs algébriques et des mesures de base suivantes :

- $P \{ \langle condition_logique \rangle \}$: correspond à la probabilité de $\langle condition_logique \rangle$.
- $P \{ \langle condition_logique1 \rangle \text{ IF } \langle condition_logique2 \rangle \}$: calcule la probabilité de $\langle condition_logique1 \rangle$ sachant que $\langle condition_logique2 \rangle$.
- $E \{ \langle fonction_marque(p) \rangle \}$: renvoie le nombre de jetons dans une place en exécutant la fonction : $\langle fonction_marque(p) \rangle$.

- $E \{ \langle fonction_marque(p) \rangle \text{ IF } \langle condition_logique \rangle \}$: renvoie le nombre de jetons dans une place en exécutant la fonction : $\langle fonction_marque(p) \rangle$ et en satisfaisant la condition : $\langle condition_logique \rangle$.

Exemple de mesures :

MEASURE debit

$E\{\#Msg_libres\} * taux_arrive;$

MEASURE no

$E\{\#Orbite\};$

MEASURE ns

$E\{\#Msg_en_envoi\};$

MEASURE nd

$E\{\#Ser_oisif_oper\};$

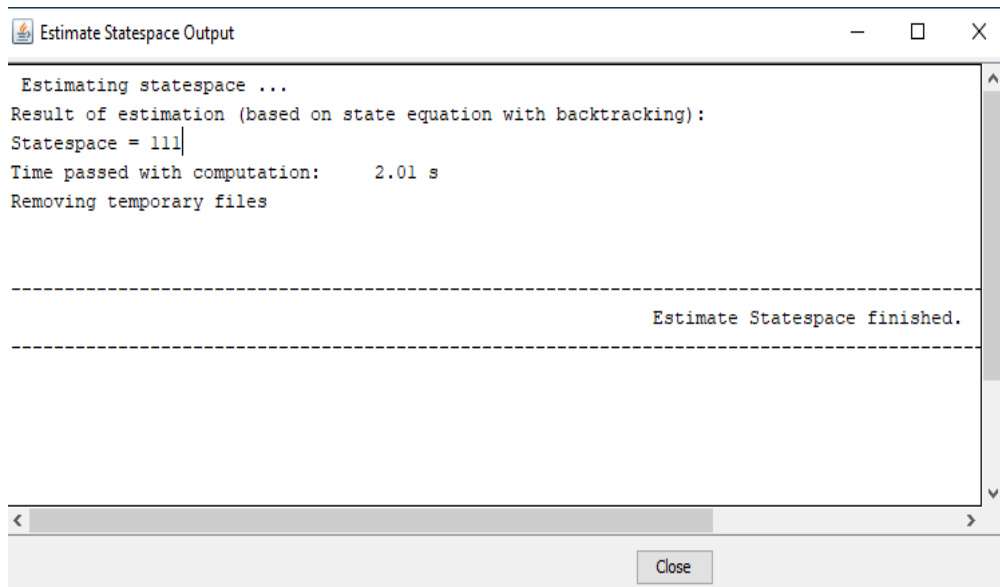
VI.3.1. Analyse qualitative du TimeNet

Le *TimeNet* nous permet d'effectuer aussi bien une analyse qualitative du modèle saisi qu'une analyse quantitative.

Sous le menu 'Validate' de l'outil TimeNet nous trouvons des fonctions basées sur la structure du modèle de RDPSG. Nous trouvons les fonctions suivantes :

- **Estimate Statespace** : calcule une estimation de la taille de l'espace d'états. La Figure VI-2 montre un exemple d'utilisation de cette fonction.
- **Traps** : calcule l'ensemble minimum de Traps (i.e. ensemble de place qui pourrait toujours être marquée). La Figure VI-3 montre un exemple d'utilisation de cette fonction.
- **Siphons** : calcule l'ensemble minimum de places absorbantes (i.e. ensemble de places qui ne pourrait être jamais marquées). La Figure VI-4 montre un exemple d'utilisation de cette fonction.

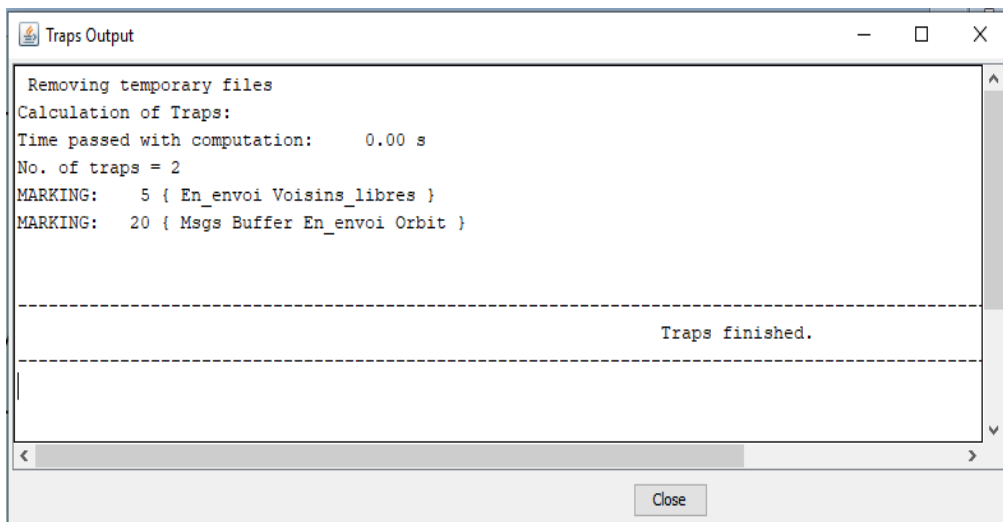
- **Check Structure** : permet d'obtenir les invariants du modèle et les conflits des transitions immédiates (la Figure VI-5 montre un exemple d'utilisation de cette fonction).



```
Estimate Statespace Output
Estimating statespace ...
Result of estimation (based on state equation with backtracking):
Statespace = 111
Time passed with computation:    2.01 s
Removing temporary files

-----
Estimate Statespace finished.
-----
```

Figure VI-2 Résultat de l'estimation de la taille de l'espace d'états



```
Traps Output
Removing temporary files
Calculation of Traps:
Time passed with computation:    0.00 s
No. of traps = 2
MARKING:    5 { En_envoi Voisins_libres }
MARKING:    20 { Msgs Buffer En_envoi Orbit }

-----
Traps finished.
-----
```

Figure VI-3 Résultat de l'exécution de la fonction *Traps*

```

Siphons Output
Removing temporary files
Calculation of Siphons:
Time passed with computation: 0.00 s
No. of siphons = 2
MARKING: 5 { En_envoi Voisins_libres }
MARKING: 20 { Msgs Buffer En_envoi Orbit }

-----
Siphons finished.
-----
Close

```

Figure VI-4 Résultat de l'exécution de la fonction *Siphons*

```

Structural Analysis Output
STRUCTURAL ANALYSIS ...
The net contains 2 P-invariants.
#En_envoi + #Voisins_libres = s
#Msgs + #Buffer + #En_envoi + #Orbit = N
Uncovered places: Battery Standby
The model contains mark-dependent transitions. T-invariants calculation are not execute.
EXTENDED CONFLICT SET
Priority immediate Transitions
1 debut_envoi
1 Vers_orbit
Removing temporary files

-----
Structural Analysis finished.
-----
Close

```

Figure VI-5 Résultat de l'exécution de la fonction *Check Structure*

VI.3.2. Les méthodes d'analyse du *TimeNet*

Le menu **Evaluation** de l'interface graphique nous permet d'accéder aux fonctions d'évaluation de performance du modèle. Plusieurs méthodes d'évaluation sont disponibles (Figure VI-6), nous citons :

- L'analyse stationnaire.
- La simulation stationnaire.
- L'analyse transitoire.
- La simulation transitoire.

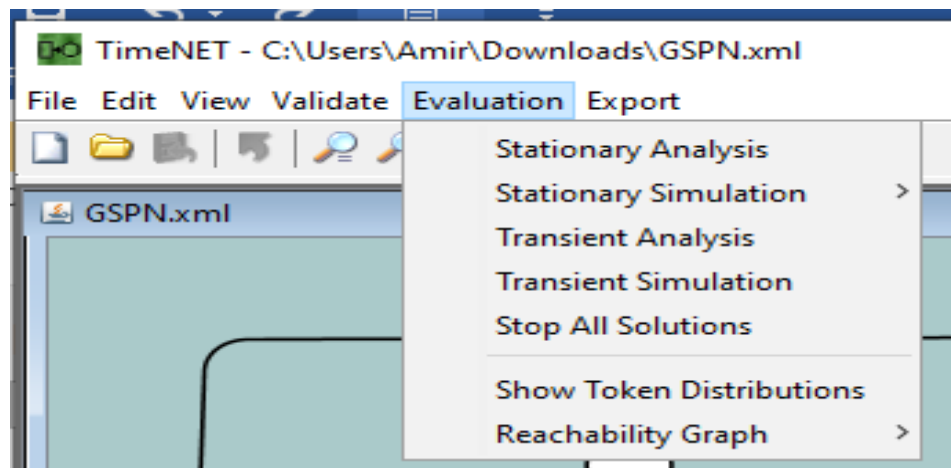


Figure VI-6 Menu d'évaluation du *TimeNet*

Nous présentons ici l'analyse en régime stationnaire

VI.3.3. Analyse stationnaire

L'analyse en régime stationnaire permet de calculer la solution d'un modèle (la Figure VI-7 montre la fenêtre du *TimeNet* spécifique à cette analyse).

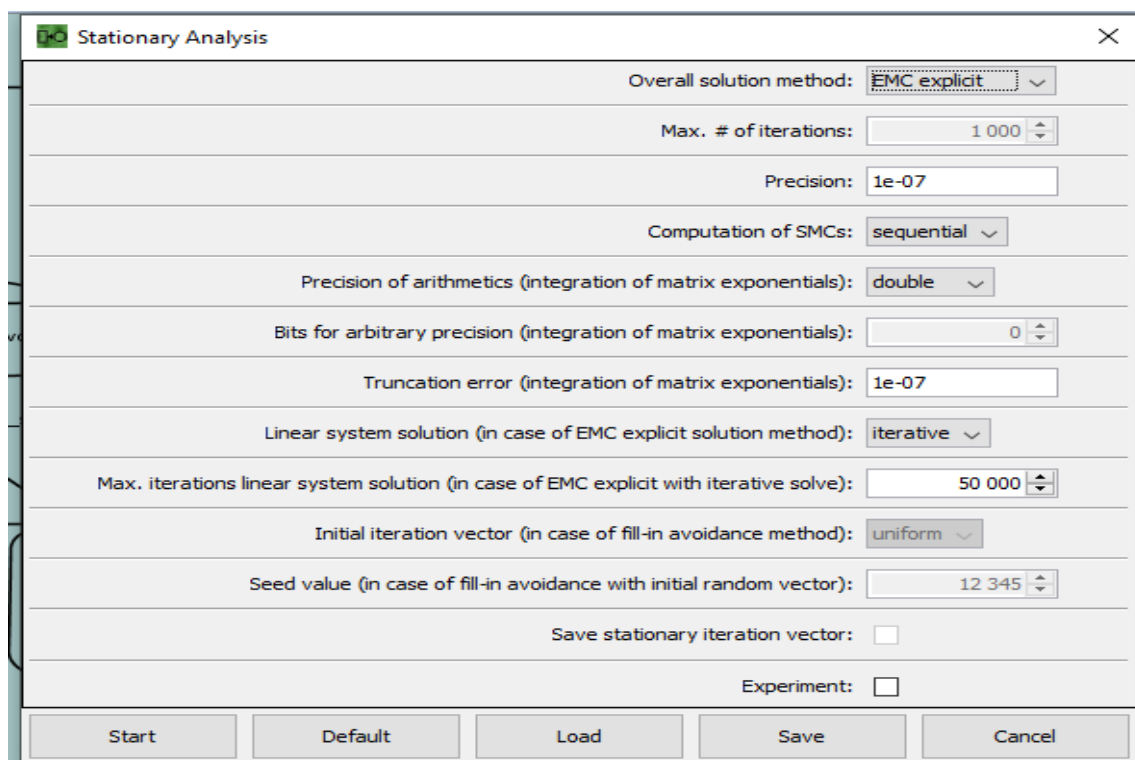


Figure VI-7 Fenêtre permet une analyse stationnaire

L'analyse stationnaire du *TimeNet* permet d'obtenir un ensemble de données qui peuvent être utilisés pour construire des graphes (la Figure VI-8 représente les options de l'exécution expérimentale).

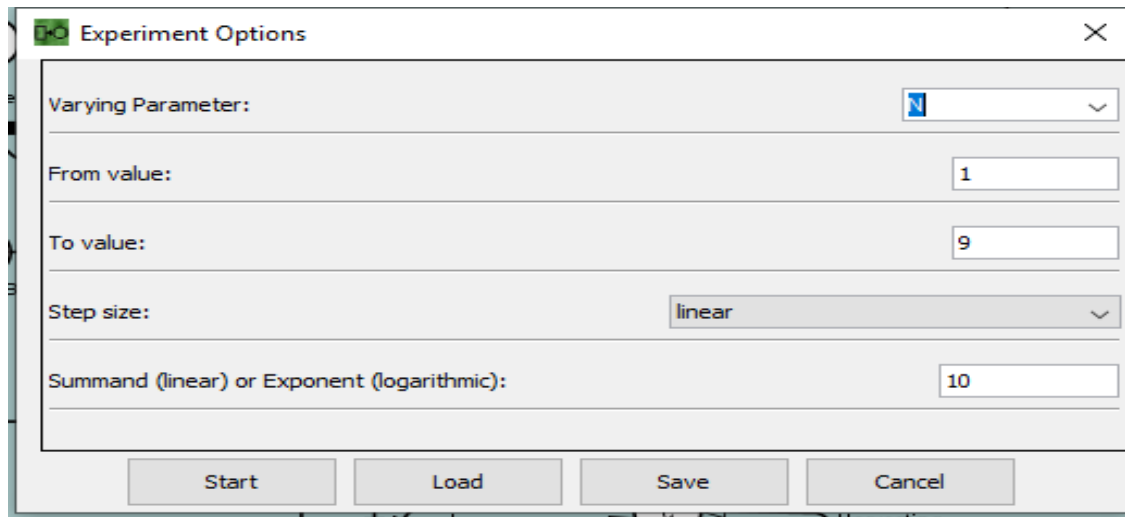


Figure VI-8 Fenêtre permet une exécution expérimentale

VI.3.4. Analyse numérique des résultats

- **La charge moyenne de la batterie par rapport au taux_harvesting** (la figure VI-9 montre l'influence du taux_harvesting sur la charge moyenne de la batterie).

La charge moyenne de la batterie augmente lorsque le *taux_harvesting* augmente. Ce résultat est logique et coïncide avec la réalité. La charge moyenne de la batterie se stabilise près de 99% avec cette configuration.

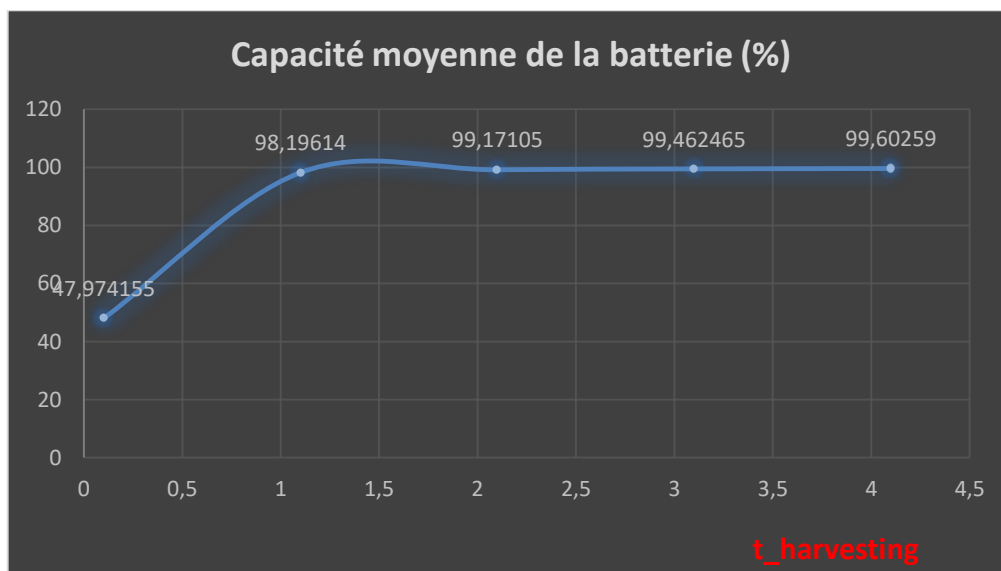


Figure VI-9 l'influence du taux_harvesting sur la charge moyenne de la batterie.

- **Le temps de réponse moyen par rapport au *taux_sleep*** (la figure VI-10 représente le temps de réponse moyen par rapport au *taux_sleep*).

Le temps de réponse moyen augmente lorsque le *taux_sleep* augmente. Lorsque le SN est activé dans la plupart des cas, il sert des messages dès qu'ils ont reçu.

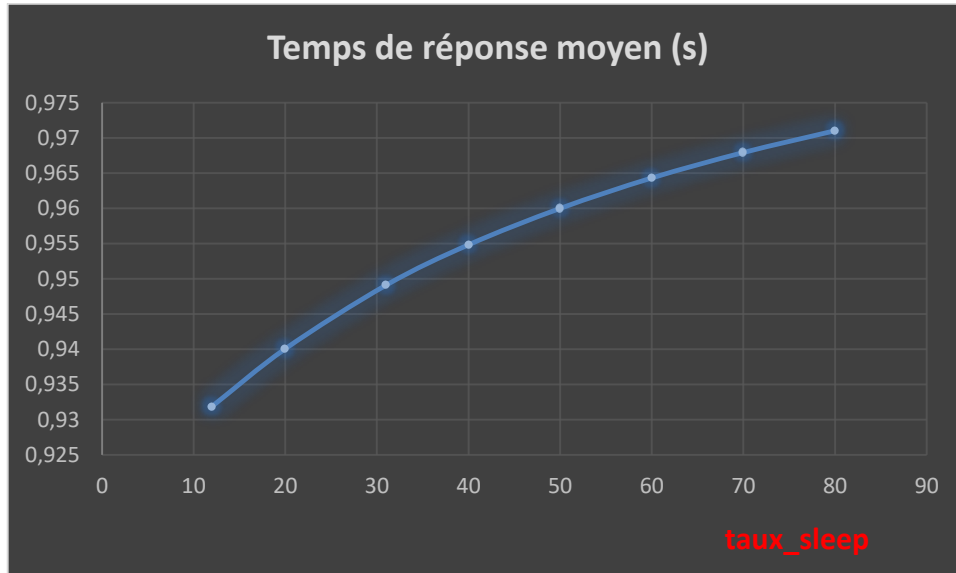


Figure VI-10 le temps de réponse moyen par rapport au *taux_sleep*.

- **Le temps de réponse moyen par rapport au *taux_harvesting*** (la figure VI-11 représente le temps de réponse moyen par rapport au *taux_harvesting*).

Le temps de réponse moyen augmente lorsque le *taux_harvesting* augmente jusqu'à la valeur de 1, ensuite il diminue quand le taux de harvesting augmente. Ce résultat est clair, puisque l'énergie a une influence sur la transmission

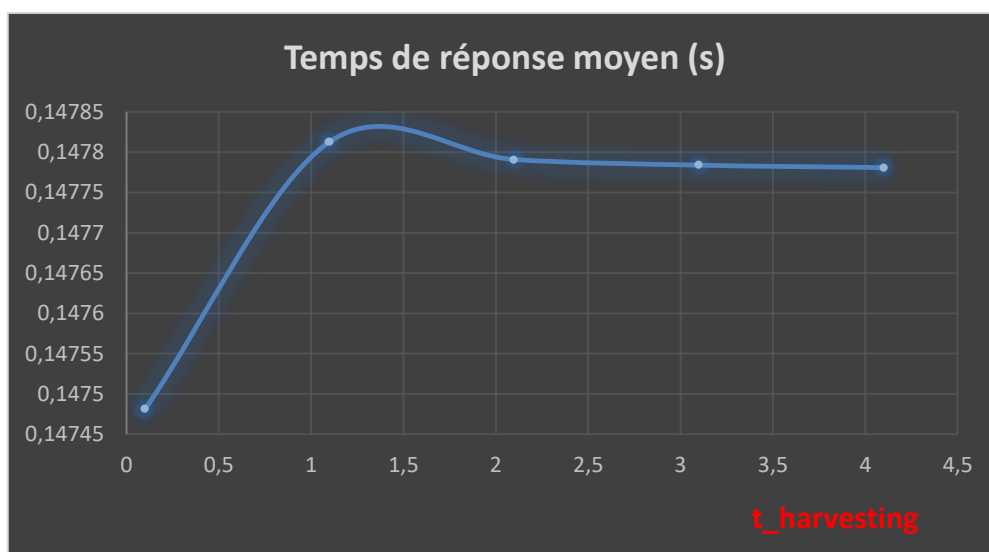


Figure VI-11 le temps de réponse moyen par rapport au *taux_harvesting*.

- **La charge moyenne de la batterie par rapport au taux de sommeil** (la figure VI-12 montre l'influence du *taux_sleep* sur la moyenne charge de la batterie).

La charge moyenne de la batterie augmente lorsque le taux de sommeil augmente. Le SN conserve plus d'énergie lorsqu'il est dans l'état de sommeil (vielle).

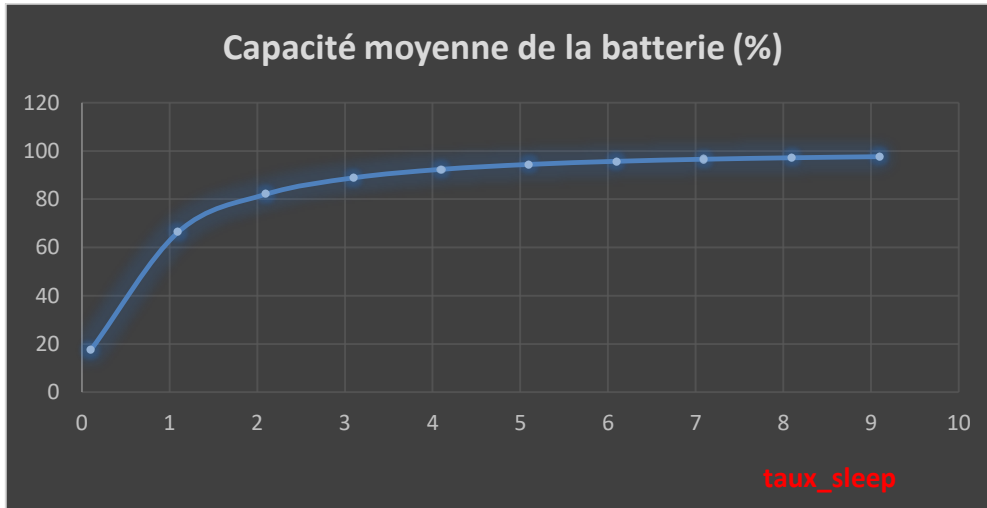


Figure VI-12 l'influence du *taux_sleep* sur la charge moyenne de la batterie.

- **Le temps de réponse moyen par rapport au nombres messages envoyés(N)** (la figure VI-13 représente le temps de réponse moyen par rapport au nombre_msg N.)

Le temps de réponse moyen augment lorsque les messages à envoyer augmentent.

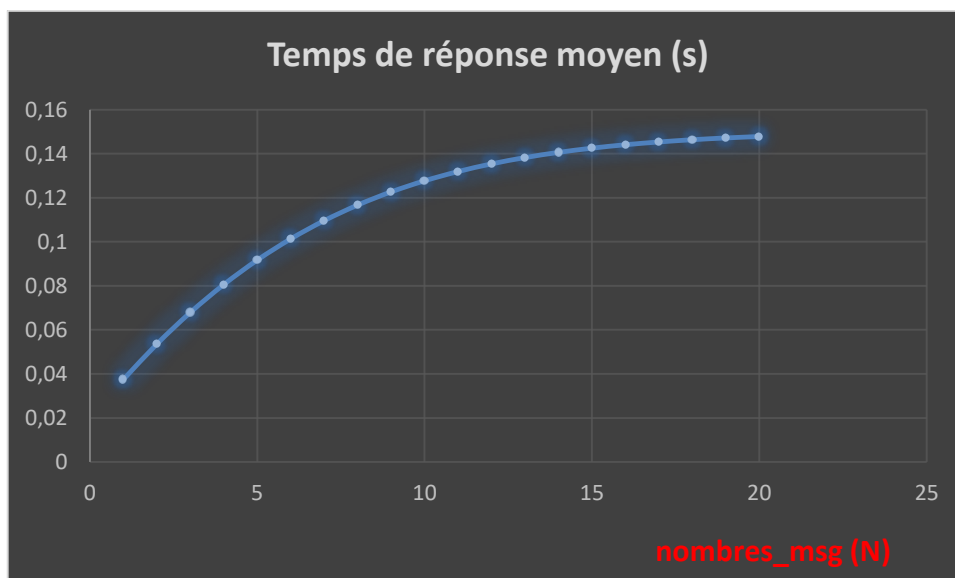


Figure VI-13 temps de réponse moyen par rapport au nombre_msg N.

- **Le temps de réponse moyen par rapport au capacité de la batterie « C »** (la figure VI-14 représente le temps de réponse moyen par rapport au capacité de batterie C).

Le temps de réponse moyen diminue lorsque la capacité de la batterie *augmente*.

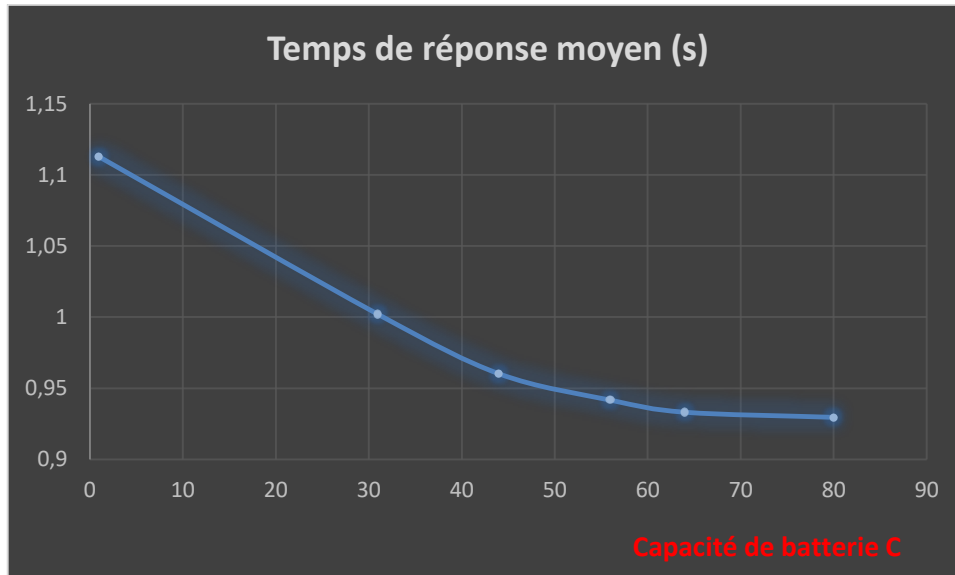


Figure VI-14 le temps de réponse moyen par rapport au capacité de la batterie C

VI.4. Conclusion

Dans ce chapitre, nous avons présenté un modèle GSPN pour les réseaux de capteurs sans fil avec récupération d'énergie et mode de veille.

La batterie d'un capteur est représentée en quantums. Après l'étude d'ergodicité nous avons donné quelques analyses sur les paramètres de performances essentiels : la charge moyenne de la batterie et le temps de réponse moyenne pour envoyer un message.

L'analyse expérimentale est donnée pour montrer la capacité de notre modèle à ajuster les valeurs d'entrée pour obtenir un comportement souhaitable du réseau.

Conclusion générale et perspectives

Dans notre travail, nous sommes intéressés à la problématique de la conservation de l'énergie dans un réseau de capteurs sans fil. Dans ce dernier, les nœuds capteurs sont alimentés par des batteries, généralement irremplaçables car les nœuds capteurs sont déployés dans des zones difficilement accessibles. Afin de prolonger la durée de vie du réseau de capteurs en minimisant la consommation d'énergie, plusieurs solutions ont été proposées. La majorité de ces solutions essayent d'éviter les différentes causes de perte d'énergie. Généralement, ces solutions ne sont pas suffisamment optimales ce qui laisse l'énergie dans un réseau de capteur un problème de recherche ouvert.

Notre contribution dans ce mémoire consiste à proposer une solution qui se base sur un *mécanisme de veille* et utilise la technique d' *Energy Harvesting* pour minimiser la consommation d'énergie dans le réseau et utiliser l'énergie de l'environnement pour charger la batterie à chaque fois quand la capacité de la batterie diminue.

L'utilisation de la technique de veille consiste à utiliser énergie seulement quand il y a un travail sinon le capteur est mis en veille c'est-à-dire l'arrêt de la consommation d'énergie quand il n'existe pas un travail et conserve en conséquence l'énergie du réseau. Les résultats de simulation montrent que notre solution est efficace en termes de consommation d'énergie et prolonge le plus longtemps possible la durée de vie du réseau.

L'utilisation de technique d'énergie *Harvesting* pour récupérer l'énergie renouvelable de l'environnement pour charger la batterie donc augmenter la durée de vie de ce réseau sans fil est efficace.

Notre solution fait appel à des opérations consommatrices d'énergie telles que ces deux mécanismes sont bien évalués avec une analyse de performances sur notre modèle GSPN. Nous avons obtenu des résultats positifs sur la capacité de la batterie. Elle augmente à chaque fois le taux de veille augmente et aussi augmente à chaque fois le taux de récupération d'énergie augmente (*Harvesting*) c'est-à-dire que nos mécanismes sont efficaces.

Dans un travail futur, nous envisageons d'optimiser ces opérations pour minimiser davantage la consommation d'énergie dans le réseau et considérer autres circonstances tel que : La perte de messages, les pannes et autres.

Bibliographie

1. Akyildiz, I.F., et al., *A survey on sensor networks*. IEEE Communications magazine, 2002. **40**(8): p. 102-114.
2. Karl, H. and A. Willig, *A short survey of wireless sensor networks*. Telecommunication Networks Group, Technische Universität Berlin, Hasso-Plattner Institute, Potsdam, 2003.
3. Swami, A., et al., *Wireless sensor networks: signal processing and communications perspectives*. 2007: John Wiley & Sons.
4. Chávez, E., N. Mitton, and H. Tejada. *Routing in wireless networks with position trees*. in *International Conference on Ad-Hoc Networks and Wireless*. 2007. Springer.
5. Yang, K., *A study on power-friendly routing protocols for sensor networks*. 2005, Technical Report, University of Essex Department of Electronic Systems
6. Raghunathan, V., et al., *Energy-aware wireless microsensor networks*. IEEE Signal processing magazine, 2002. **19**(2): p. 40-50.
7. Vieira, M.A.M., et al. *Survey on wireless sensor network devices*. in *EFTA 2003. 2003 IEEE Conference on Emerging Technologies and Factory Automation. Proceedings (Cat. No. 03TH8696)*. 2003. IEEE.
8. Al-Karaki, J.N. and A.E. Kamal, *Routing techniques in wireless sensor networks: a survey*. IEEE wireless communications, 2004. **11**(6): p. 6-28.
9. Sahoo, B., S. Rath, and D. Puthal, *Energy efficient protocols for wireless sensor networks: A survey and approach*. International Journal of Computer Applications, 2012. **44**(18): p. 43-48.
10. Santi, P., *Topology control in wireless ad hoc and sensor networks*. ACM computing surveys (CSUR), 2005. **37**(2): p. 164-194.
11. Meguerdichian, S., et al. *Exposure in wireless ad-hoc sensor networks*. in *Proceedings of the 7th annual international conference on Mobile computing and networking*. 2001. ACM.
12. Akyildiz, I.F. and M.C. Vuran, *Wireless sensor networks*. Vol. 4. 2010: John Wiley & Sons.
13. Shih, E., et al. *Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks*. in *Proceedings of the 7th annual international conference on Mobile computing and networking*. 2001. ACM.
14. Yoneki, E. and J. Bacon, *A survey of Wireless Sensor Network technologies*. UCAM-CL-TR-646, 2005.
15. Heinzelman, W.R., A. Chandrakasan, and H. Balakrishnan. *Energy-efficient communication protocol for wireless microsensor networks*. in *Proceedings of the 33rd annual Hawaii international conference on system sciences*. 2000. IEEE.
16. Khelladi, L. and N. Badache, *Les réseaux de capteurs: état de l'art*. Faculté électronique et informatique Bab Ezzouar-Algérie, 2004.
17. Salunke, A. and D. Ambawade. *Dynamic Sequence Number Thresholding protocol for detection of blackhole attack in Wireless Sensor Network*. in *2015 International Conference on Communication, Information & Computing Technology (ICCICT)*. 2015. IEEE.
18. Castelluccia, C. and A. Francillon, *Protéger les réseaux de capteurs sans fil*. SSTIC08, 2008.
19. Bakli, N.T.L., *"Etude et amélioration d'un protocole de routage efficace en énergie dans les réseaux de capteurs"*. Mémoire fin d'étude faculté des sciences et des sciences de l'ingénieur. 2011.
20. Hill, J., et al. *System architecture directions for networked sensors*. in *ACM SIGOPS operating systems review*. 2000. ACM.
21. Levis, P.A., D.E. Gay, and D.E. Culler, *Bridging the gap: Programming sensor networks with application specific virtual machines*. 2004: Citeseer.
22. Manjeshwar, A. and D.P. Agrawal. *TEEN: ARouting Protocol for Enhanced Efficiency in Wireless Sensor Networks*. in *ipdps*. 2001.
23. Zhao, Q. and L. Tong. *Distributed opportunistic transmission for wireless sensor networks*. in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*. 2004. IEEE.
24. *WirelessSensor-Networks*. Available from: <http://www.xbow.com/Products/WirelessSensor-Networks>.
25. Rajaravivarma, V., Y. Yang, and T. Yang. *An overview of wireless sensor network and applications*. in *Proceedings of the 35th Southeastern Symposium on System Theory, 2003*. 2003. IEEE.
26. Deng, J., et al., *Scheduling sleeping nodes in high density cluster-based sensor networks*. Mobile Networks and Applications, 2005. **10**(6): p. 825-835.
27. Nieberg, T., et al., *Collaborative algorithms for communication in wireless sensor networks*, in *Ambient Intelligence: Impact on Embedded Sytem Design*. 2003, Springer. p. 271-294.
28. Younis, M. and T. Nadeem, *Energy efficient MAC protocols for wireless sensor networks*. Technical report, 2004.
29. Ilyas, M. and I. Mahgoub, *Handbook of sensor networks: compact wireless and wired sensing systems*. 2004: CRC press.
30. Namgoong, H., D. Lee, and D. Nam. *Energy efficient topology for wireless microsensor networks*. in *Proceedings of the 2nd ACM international workshop on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks*. 2005. ACM.

31. Hentati, A., *Sélection des nœuds dans les réseaux de capteurs sans fil avec récolte d'énergie*. 2016, École Polytechnique de Montréal.
32. Basagni, S., et al., *Mobile ad hoc networking: Cutting edge directions*. Vol. 35. 2013: John Wiley & Sons.
33. Petri, C.A. and W. Reisig, *Petri net*. Scholarpedia, 2008. **3**(4): p. 6477.
34. Moehlis, J., K. Josic, and E.T. Shea-Brown, *Periodic orbit*. Scholarpedia, 2006. **1**(7): p. 1358.
35. Pikovsky, A., et al., *Synchronization: a universal concept in nonlinear sciences*. Vol. 12. 2003: Cambridge university press.
36. Marsan, M.A., et al., *Modelling with generalized stochastic Petri nets*. Vol. 292. 1995: Wiley New York.
37. Silva, M. and E. Teruel, *Petri nets for the design and operation of manufacturing systems*. European journal of control, 1997. **3**(3): p. 182-199.
38. Silva, M., et al. *Petri nets and production systems*. in *Advanced Course on Petri Nets*. 1996. Springer.
39. Tuffin, B. and K.S. Trivedi. *Implementation of importance splitting techniques in stochastic Petri net package*. in *International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*. 2000. Springer.
40. Haverkort, B.R., H.C. Bohnenkamp, and C.U. Smith, *Computer Performance Evaluation. Modelling Techniques and Tools: 11th International Conference, TOOLS 2000 Schaumburg, IL, USA, March 25-31, 2000 Proceedings*. 2000: Springer Science & Business Media.
41. Van der Aalst, W.M., K.M. Van Hee, and H.A. Reijers, *Analysis of discrete-time stochastic petri nets*. Statistica Neerlandica, 2000. **54**(2): p. 237-255.
42. Vautherin, J. *Parallel systems specifications with coloured Petri nets and algebraic specifications*. in *European Workshop on Applications and Theory in Petri Nets*. 1986. Springer.
43. Marsan, M.A. *Stochastic Petri nets: an elementary introduction*. in *European Workshop on Applications and Theory in Petri Nets*. 1988. Springer.
44. Bobbio, A., et al., *Recent developments in non-Markovian stochastic Petri nets*. Journal of Circuits, Systems, and Computers, 1998. **8**(01): p. 119-158.
45. Ioualalen, M., *Evaluation des performances et de la fiabilité des systèmes multi-classes avec rappel à l'aide des réseaux de petri stochastiques colorés*. 2007.
46. Marsan, M.A., et al., *The effect of execution policies on the semantics and analysis of stochastic Petri nets*. IEEE Transactions on Software Engineering, 1989. **15**(7): p. 832-846.
47. Diaz, M., *Les réseaux de Petri: modèles fondamentaux*. 2001: Hermes.
48. Dutheillet Lamonthézie, C., *Symetries dans les reseaux colores: definition, analyse et application a l'evaluation de performance*. 1991, Paris 6.
49. He, Q., Z. Li, and X. Zhang. *Study on cloud storage system based on distributed storage systems*. in *2010 International Conference on Computational and Information Sciences*. 2010. IEEE.
50. Zimmermann, 2008.
51. Rashid, A., et al., *Improving Energy Conservation in Wireless Sensor Networks using Energy Harvesting System*. International Journal of Advanced Computer Science and Applications (IJACSA), 2018. **9**: p. 354-361.
52. Florin, G., C. Fraize, and S. Natkin, *Stochastic Petri nets: Properties, applications and tools*. Microelectronics Reliability, 1991. **31**(4): p. 669-697.
53. Peterson, J.L., *Petri net theory and the modeling of systems*. 1981: Prentice-hall.
54. Wang, J., J. Cao, and Q. Li, *Reliability analysis of the retrieval queue with server breakdowns and repairs*. Queueing Systems, 2001. **38**(4): p. 363-380.