

Table des matières

table des Figures	iii
1 Description des réseaux de capteurs sans fil	2
1.1 Introduction	2
1.2 Motivations et Enjeux	3
1.2.1 Motivations	3
1.2.2 En jeux	3
1.3 les capteurs	3
1.3.1 Définition	3
1.3.2 Classification d'un RCSF	4
1.4 Architecture d'un nœud de capteur	5
1.4.1 Noeud Source	6
1.4.2 noeud puits	6
1.5 Généralité sur les réseaux de capteurs sans fil	6
1.5.1 Définition	6
1.5.2 Caractéristiques d'un RCSF	7
1.5.3 Architecture d'un réseau de capteurs sans fil	8
1.5.4 Domaine d'application des RCSF	9
1.5.5 Contraintes de conception des RCSF	9
1.6 Points clés pour la consommation des RCSF	11
1.7 Classification des réseaux de capteurs sans fil	11
1.7.1 Selon le mode de communication	11
1.8 Le routage dans le RCSF	12
1.8.1 Définition de routage	12
1.8.2 Classification des protocoles de routage pour les RCSFs	12
1.9 Quelques définitions cruciales	15
1.10 Conclusion	16
2 Quelques rappels sur la théorie des graphes	17
2.1 introduction	17
2.2 Concepts de base	18
2.2.1 Définitions générales	18
2.2.2 Graphe orienté	18
2.2.3 Graphe non orienté	18
2.2.4 Arcs adjacents, arêtes adjacentes	19
2.3 Connexité dans les graphes	19
2.3.1 chaine	19

2.3.2	Chemin	19
2.3.3	Cycle	19
2.3.4	Circuit	20
2.3.5	connexité	20
2.3.6	Forte connexité	20
2.3.7	Arbres et forêts	20
2.4	Quelques types de graphes	20
2.4.1	Graphe valué	20
2.4.2	Graphe Complet	21
2.4.3	Sous-graphe	21
2.4.4	Sous-graphe partiel	21
2.4.5	graphe planaire	21
2.4.6	Graphe biparti	22
2.5	Plus courts chemins	22
2.5.1	Problème du plus court chemin	22
2.6	Algorithme de recherches de Plus courts chemins	23
2.7	Conclusion	23
3	Protocole de routage basée sur d'algorithme de Dijkstra dans un RCSFs	24
3.1	introduction	24
3.2	Modèle du réseau	24
3.3	Solution proposée	24
3.4	Hypothèses	25
3.5	Détail de la solution proposée	25
3.5.1	Objectif de notre modélisation	25
3.6	Application sur le C++	32
3.6.1	Présentation de langage C++	32
3.6.2	Application numérique	32
3.7	Conclusion	38
3.8	Conclusion générale	39

Table des figures

1.1	schéma d'un capteur sans fil.	4
1.2	Architecture d'un capteur.	5
1.3	Nœuds capteurs dispersés dans un champ de capteur	7
1.4	architecture d'un réseau de capteur sans fil	8
1.5	les types de communication dans les réseaux de capteurs sans fil	12
1.6	Classification des protocoles de routage	13
1.7	Protocoles de routage à plat.	13
1.8	Topologie hiérarchique.	14
2.1	les sept ponts de Königsberg	17
2.2	Graphe orienté	18
2.3	Graphe non orienté	19
2.4	Graphe complet	21
2.5	Sous-graphe, graphe partiel	21
2.6	Graphe planaire	22
2.7	Graphe biparti	22
3.1	Réseau associé après la modélisation	26
3.2	Réseau obtenu après l'initialisation	26
3.3	Réseau obtenu à la première itération	27
3.4	Réseau obtenu à la deuxième itération	28
3.5	Réseau obtenu à la troisième itération	28
3.6	Réseau obtenu à la quatrième itération	29
3.7	Réseau obtenu à la cinquième itération	30
3.8	Réseau obtenu à la sixième itération	30
3.9	Réseau obtenu à la septième itération	31
3.10	Introduire les données	33
3.11	Introduire les données	34
3.12	Introduire les données	34
3.13	Introduire les données	35
3.14	Résultat	36
3.15	Résultat	37
3.16	Résultat	38

Introduction générale

Durant ces dernières décennies, les réseaux de capteurs ont bouleversé le monde. Le besoin d'un suivi continu des phénomènes naturels et aussi la surveillance dans différents domaines, ont renforcé l'intérêt pour cette nouvelle ère de l'informatique embarquée. En revanche, les réseaux de capteurs souffrent de leurs fragilités et de leurs énergies limitées. Les noeuds capteurs sont alimentés par des batteries limitées en énergie. Par ailleurs, le remplacement des batteries n'est pas une solution envisageable pour ces derniers, soit à cause du déploiement aléatoire des capteurs, ou à cause de l'hostilité de l'environnement où ils sont placés. Toutefois, la mort d'un ou plusieurs noeuds capteurs interrompt partiellement la communication dans le réseau. De ce fait, une partie des données collectées sera perdue ce qui en résulte à la mort partielle du réseau[14].

Dans ce mémoire, nous intéressons à résoudre un problème réel en utilisant la théorie de graphe et plus particulièrement le problème de plus court chemin dans les réseaux, L'objectif principal c'est l'utilisation de l'algorithme de Dijkstra pour trouver des techniques de routage efficaces en termes d'énergie afin que la durée de vie du réseau soit maximisée. Ce mémoire est organisé en trois chapitres comme suit :

.Le premier chapitre nous a permis d'avoir une vue globale sur les réseaux de capteurs sans fil, leurs architectures, leurs caractéristiques, leurs domaines d'application, les réseaux de capteurs sans fil représentent un intérêt considérable vu à l'évolution connue récemment dans ce domaine. Par conséquent, nous avons remarqué que ces réseaux ont plusieurs contraintes, dont le problème majeur est la consommation d'énergie.

Sachant que la transmission directe (un seul saut) vers la station de base épuise beaucoup plus d'énergie qu'une transmission multi-saut qui nous donne un gain considérable en énergie ainsi prolonger la durée de vie du réseau et c'est sur ce contexte que nous allons présenter un état de l'art sur le routage afin d'effectuer une transmission multi-saut.

.l'objet de deuxième chapitre présenter quelques définitions de base sur la théorie des graphes et Algorithme de recherches de Plus courts chemins qui seront utilisées dans la suite de notre travail.

. Ce dernier chapitre est consacré à la modélisation d'un réseau de capteur sans fil, sachant que plus le noeud source s'éloigne du noeud puit, plus de consommation d'énergie, et dans le contexte de l'étude de l'énergie on propose un protocole de routage basée sur l'algorithme dijkstra et la programmation de ce dernier sur le langage C++ pour trouver le meilleur chemin ayant un poids minimal, entre la source et la station de base.

Chapitre 1

Description des réseaux de capteurs sans fil

1.1 Introduction

Les réseaux de capteurs sans fils (RCSF) sont des réseaux ad hoc généralement constitués d'entité autonome miniaturisés appelés nœuds capteurs pouvant communiquer par liaison radio. Les RCSF ont suscités beaucoup d'engouements dans la recherche scientifique en raison notamment des nouveaux problèmes de routage sous forte contrainte de durée de vie du réseau et de faible capacité des nœuds. [8]

Dans ce qui suit nous allons définir un capteur, son architecture ensuite on présentera des généralités sur les RCSF à savoir l'architecture des RCSF, les caractéristiques, et domaines d'applications enfin on termine par la description du routage, ses types et le protocole de communication.

1.2 Motivations et Enjeux

1.2.1 Motivations

Les avancées technologiques en matière de production de circuits électroniques, et dans Une moindre mesure, les progrès dans le domaine des batteries pour le stockage de l'énergie électrique, ont permis le développement de capteurs autonomes, de petite taille et dont le Coût de fabrication est de plus en plus faible. Les capteurs sont des appareils capables de Récolter différentes sortes d'informations dans leur rayon d'action et de communiquer avec D'autres capteurs. Les capteurs sont par conséquent des appareils peu complexes et faciles à configurer. Leur faible coût permet ainsi de déployer un grand nombre de capteurs afin de surveiller de Vastes zones accidentées ou inaccessibles. Grâce au grand nombre de capteurs qui le constitue, Un réseau de capteurs sans fil résiste plus facilement aux attaques d'un ennemi ou aux Dysfonctionnements spontanés, ce qui le rend très appropriée aux applications militaires [1].

La technologie des RCSF est un domaine de recherche nouveau, par rapport a la technologie d'internet. Cette technologie des RCSF a bénéficié d'une position centrale dans l'espace de recherche des réseau émergents futures ces dernières années. il y'a de plus en plus de travaux de recherche intéressants sur plusieurs aspects des RCSF : énergie, localisation, synchronisation, mobilité, et changement de topologie, qualité de service, sécurité, traitement dans le réseau. . . etc. mais l'axe de recherche qui a suscité le plus d'intérêt de la part de la communauté des chercheurs , jusqu'à ce jour, est celui de l'économie d'énergie[1].

1.2.2 En jeux

Dans cette section, nous décrivons deux enjeux fondamentaux dans les réseaux de capteurs : le routage et la structuration des réseaux. Le routage permet l'acheminement des informations vers une destination donnée à travers un réseau de connexion. En effet, le rôle des techniques de routage consiste à déterminer un acheminement optimal des paquets à travers le réseau au sens d'un certain critère de performance comme la consommation énergétique. Le but est de trouver l'investissement de moindre coût qui assure le routage du trafic nominal et garantit la qualité de service. Le problème qui se pose dans le contexte des réseaux de capteurs est l'adaptation de la méthode d'acheminement utilisée avec le grand nombre de nœuds existant dans un environnement caractérisé par de changements de topologies, de modestes capacités de calcul, de sauvegarde, et d'énergie. Toute conception de protocole de routage implique l'étude des problèmes suivants :

1.3 les capteurs

1.3.1 Définition

Un capteur sans fil est un petit dispositif électronique, peu coûteux, doté de ressources limitées en énergie (batterie), en puissance de calcul et en capacité de stockage. Un capteur possède la capacité de mesurer une valeur physique environnementale telle que la température, la lumière, la pression, etc. et de la transmettre en utilisant les communications sans fil à un centre de contrôle

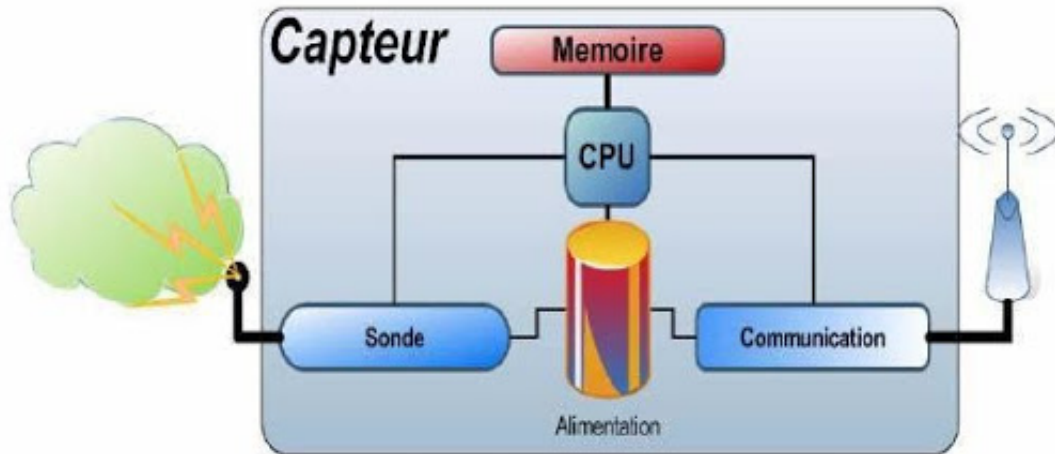


FIGURE 1.1 – schéma d'un capteur sans fil.[6]

appelé une station de base. Les nœuds capteurs peuvent être déployés dans toute application dont l'objectif est de surveiller l'environnement. [7]

1.3.2 Classification d'un RCSF

Les capteurs Actifs

Généralement, un capteur actif est un système de mesure qui nécessite une source d'énergie embarquée, la plupart du temps assurée par une batterie, et ce pour la réalisation de la phase de traitement au cours de laquelle le signal est filtré (nettoyé), amplifié et converti dans un format compatible et exploitable. Dans ce cas, le capteur doit non seulement mesurer des propriétés physiques mais doit également effectuer des tâches additionnelles au travers de circuits de traitement et de communication intégrés. Ce type de capteur est surtout utilisé pour assurer des mesures continues en temps réel. [6]

Les capteurs passifs

Les capteurs passifs sont des dispositifs qui ne possèdent pas de source d'énergie embarquée et présentent l'avantage d'être facilement intégrables. Ce type de capteur est utilisé dans des applications spécifiques (surveillance environnementale, des instruments de suivi spatial et aéronautique, des applications liées à la santé) qui nécessitent des unités de mesure miniatures, passives, de grande précision et fiables. L'objectif est d'assurer des mesures à distance des grandeurs physiques. Dans ce cas, deux différentes technologies peuvent être utilisées pour la transmission sans-fil de données : la transmission inductive et la transmission radio basée sur la réflexion (transpondeur passif). Dans ce qui suit, on présente brièvement les composants les plus répandus fondés sur ces types de technologies. [6]

1.4 Architecture d'un nœud de capteur

Un capteur est composé principalement d'une unité de : captage, de traitement, de transmission, de stockage et d'énergie. Des composants additionnels peuvent être ajoutés selon le domaine d'application, comme par exemple un système de localisation tels qu'un GPS (Global Positioning-System), un générateur d'énergie (exemple : cellules solaire) ou un mobilisateur lui permettant de se déplacer [10].

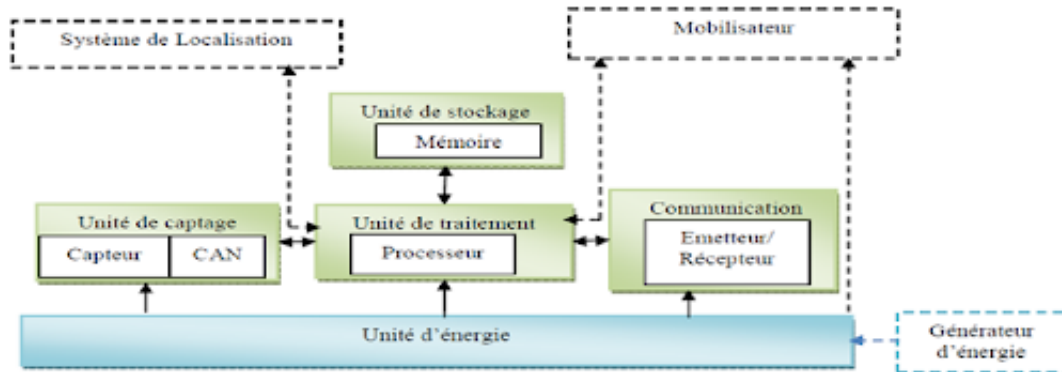


FIGURE 1.2 – Architecture d'un capteur.[10]

- **Unité d'énergie**

Un capteur est muni d'une source d'énergie, généralement une batterie, pour alimenter tous les composants. Les batteries utilisées sont soit rechargeables ou non. Souvent, dans les environnements sensibles, il est impossible de recharger ou changer une batterie. Pour cela, l'énergie est la ressource la plus précieuse puisqu'elle influe directement sur la durée de vie des capteurs et donc d'un réseau de capteurs [10].

- **Unité de captage**

La fonction principale de l'unité de captage est de capturer ou mesurer les données physiques à partir de l'objet cible. Elle est composée de deux sous-unités : le récepteur (reconnaissant la grandeur physique à capter) et le transducteur (convertissant le signal du récepteur en signal électrique). Le capteur fournit des signaux analogiques, basés sur le phénomène observé, au convertisseur Analogique /Numérique (CAN). Ce dernier transforme ces signaux en données numériques et les transmet à l'unité de traitement [10].

- **Unité de traitement**

L'unité de traitement exécute les procédures permettant au nœud de collaborer avec les autres nœuds pour effectuer la tâche assignée au réseau. Cette unité est également composée d'un processeur et d'un système d'exploitation spécifique. Elle acquiert les informations en provenance de l'unité d'acquisition et les envoie à l'unité de transmission [10].

- **Unité de transmission**

Cette unité est responsable de toutes les émissions et réceptions de données via un support de communication sans fil. Les différents choix de média de transmission incluent la Radiofréquence(RF), le Laser et l’Infrarouge [10].

- **Unité de stockage(Mémoire)**

L’unité de stockage inclut la mémoire de programme (dont les instructions sont exécutées par le processeur) et la mémoire de donnée (pour conserver des données fournies par l’unité de captage et d’autres données locales). La taille de cette mémoire est souvent limitée essentiellement par les considérations économiques et s’améliorera aussi probablement au fil des années[10].

1.4.1 Noeud Source

dont le rôle principal est de détecter les phénomènes physiques ou physiologiques se produisant dans son environnement immédiat afin de les transmettre, directement ou via multiples sauts, à un utilisateur final. C’est en fait un noeud capteur[17].

1.4.2 noeud puits

est un noeud régulier doté d’un convertisseur série connecté à une seconde unité de communication (GPRS, Wi-Fi, WiMax, etc.). La seconde unité de communication fournit une retransmission transparente des données provenant de noeuds capteurs à un utilisateur final ou d’autres réseaux comme internet[17].

1.5 Généralité sur les réseaux de capteurs sans fil

Les réseaux de capteur sans fil constituent une classe de réseaux ad hoc composés de nœuds capteurs mobile et/ou statique pouvant être déployés dans des environnements connus ou inconnus. Ces capteurs disposent d’une capacité énergétique leurs permettant de fonctionner de manière autonome et intelligente, et de communiquer via des liaisons radio selon des mécanismes et de routage établis préalablement. [5]

1.5.1 Définition

Un RCSF est un réseau composé de plusieurs nœuds de capteurs pouvant communiquer entre eux par liaison sans fil. La distribution de ces nœuds dans un champ d’observation permet d’obtenir des informations concernant un phénomène mesuré à différentes positions et en temps réel. Ces informations sont ensuite transmises à une Station de Base (BS) par liaison directe en mono-saut,

ou par l'intermédiaire des autres nœuds de capteur en multi-saut. La BS peut stocker localement les données pour acheter les mesures des capteurs. Elle peut également rendre ces mesures disponibles en ligne à d'autres utilisateurs. [4]

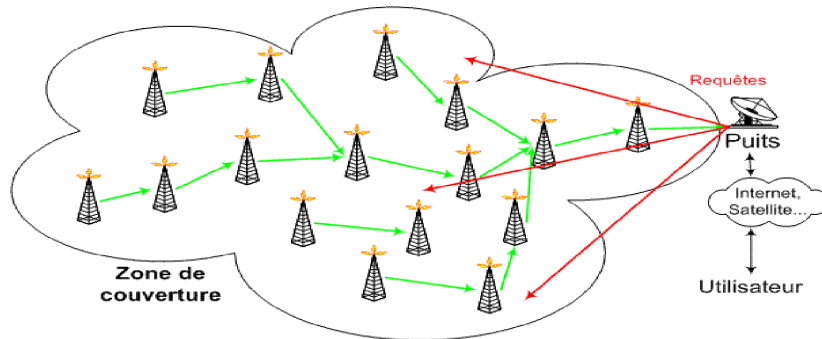


FIGURE 1.3 – Nœuds capteurs dispersés dans un champ de capteur[12]

1.5.2 Caractéristiques d'un RCSF

Un RCSF présente les caractéristiques suivantes :

- **Absence d'infrastructure** : Les réseaux ad-hoc en général, et les réseaux de capteurs en particulier se distinguent des autres réseaux par la particularité d'absence d'infrastructure préexistante et de tout genre d'administration centralisée.
- **Taille importante** : Un réseau de capteurs peut contenir des milliers de nœuds.
- **interférences** : Les liens radio ne sont pas isolés, deux transmissions simultanées sur une même fréquence, ou utilisant des fréquences proches, peuvent interférer.
- **Topologie dynamique** : Les capteurs peuvent être attachés à des objets mobiles qui se déplacent d'une façon libre et arbitraire rendant ainsi la topologie du réseau fréquemment changeante.
- **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.
- **Bande passante limitée** : Une des caractéristiques primordiales des réseaux basés sur la communication sans fil est l'utilisation d'un médium de communication partagé. Ce partage fait que la bande passante réservée à un nœud est limitée.
- **Contrainte d'énergie** : de stockage et de calcul La caractéristique la plus importante dans les réseaux de capteurs est la ressource énergétique. Car chaque capteur du réseau possède de faibles ressources en termes d'énergie (batterie). Afin de prolonger la durée de vie du

réseau, une minimisation des dépenses énergétiques est exigée chez chaque nœud du réseau. [9]

1.5.3 Architecture d'un réseau de capteurs sans fil

Un réseau de capteur sans fil (RCSF) présenté dans la Figure 1.4, est généralement constitué d'un ensemble de nœuds capteurs, variant de quelques dizaines à plusieurs milliers, déployés dans une zone géographique ou un environnement d'intérêt, communiquant entre eux par ondes radio afin d'acheminer les données captées vers un nœud collecteur appelé station de base (sink). La station de base, qui peut être un ordinateur, reçoit les données des capteurs et les transmet à l'utilisateur final par d'autres réseaux tels que Internet ou par satellite pour analyser ces données et prendre des décisions. Dans le cas le plus simple, les nœuds capteurs seront dans le voisinage direct de la station de base (communication à un saut). Cependant, dans le cas d'un réseau à grande échelle, ils ne sont pas tous dans le voisinage de la station de base et les données seront acheminées d'un nœud source vers la station de base en transitant par plusieurs nœuds, selon un mode de communication multi-sauts.

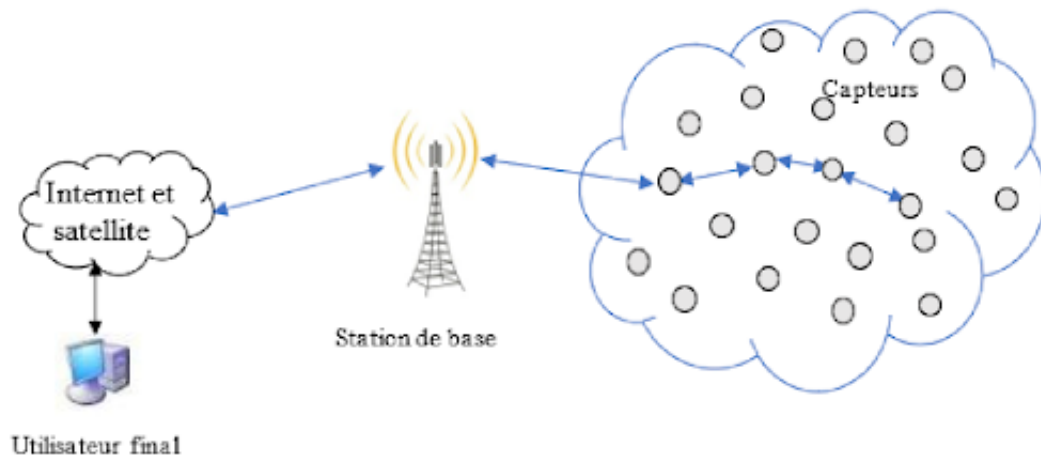


FIGURE 1.4 – architecture d'un réseau de capteur sans fil [7]

En fonction de la taille et de la topologie du réseau, il pourrait y avoir une ou plusieurs stations de bases qui peuvent être fixes dans des positions ou mobiles dans la zone du réseau. La station de base est habituellement supposée avoir une grande capacité de calcul et elle n'a pas de contraintes en termes d'énergie et de stockage. La tâche d'un nœud capteur comprend la création des paquets de données captées et la transmission sans fil de ces données à la station de base ou à d'autres nœuds capteurs. Lorsque le nœud capteur consomme toute son énergie (sa batterie est épuisée), il ne peut fournir aucun service, y compris la détection, le traitement des données ou la communication de données. Si cela se produit, le nœud capteur est considéré comme un nœud mort et il sera retiré de la topologie du réseau. [7]

1.5.4 Domaine d'application des RCSF

Grace aux évolutions de la technologie touchant les domaines : électronique, informatique industrielle, instrumentation, réseaux et télécommunication, le champ d'applications des réseaux de capteurs sans fils est de plus en plus en élargissement. Parmi les applications des RCSF nous trouvons [3] :

— **Les applications militaires**

Comme la plupart des technologies, les applications militaires étaient les premières à intégrer les RCSFs. Ils sont déployés dans un secteur stratégique ou difficile d'accès, pour y surveiller tous les mouvements (alliés ou ennemis), pour analyser le champ de bataille avant d'y envoyer du renfort, la détection des attaques biologiques ou chimiques, ... etc .

— **Les applications médicales**

Le suivi des personnes dépendantes est un véritable défi. Les RCSFs propose un système de suivi pour les personnes dépendantes vivant seules à domicile ou dans un établissement 24 h /24. Ce système est basé sur la présence d'un réseau multi-capteurs déployé dans le cadre de vie de la personne surveillée couplé à un système d'identification sans fil et un algorithme d'apprentissage.

— **Les applications environnementales**

Ces types de RCSF peuvent servir à l'observation d'un site susceptible de subir les effets d'une pollution, ou bien d'éventuel incendie, inondation, volcan ou tsunami. La construction en temps réel d'une cartographie grâce à des capteurs, disséminés sur le site capable de relever des informations sur la pollution, la température, le niveau d'eau et d'oxygène. ... , etc. Ces informations doivent être relayées vers les services spécialisés.

— **Les applications industrielles**

Dans ce domaine, les réseaux de capteurs sans fil offrent une grande flexibilité d'emploi puisqu'ils permettent de s'affranchir des contraintes liées aux câblages. Il est alors possible de satisfaire des contraintes de poids, de mobilité, de facilité de déploiement, ... etc .

Parmi les applications des réseaux de capteurs sans fil dans le milieu industriel, nous citons la surveillance de l'état de santé d'un ouvrier ou du risque de le voir exposé à des conditions de travail dangereuses (exposition à la radioactivité), la gestion des stocks, contrôle des machines à distances, etc.

1.5.5 Contraintes de conception des RCSF

Les principaux facteurs et contraintes influençant l'architecture des réseaux de Capteurs peuvent être résumés comme suit [11] :

- **La tolérance de fautes** : Certain nœuds peuvent générer des erreurs ou ne Plus fonctionner

à cause d'un manque d'énergie, un problème physique ou une interférence. Ces problèmes n'affectent pas le reste du réseau, c'est le principe de la tolérance de fautes. La tolérance de fautes est la capacité de maintenir les fonctionnalités du réseau sans interruptions dues à une erreur intervenue sur un ou plusieurs capteurs.

- **L'échelle :** Le nombre de nœuds déployés pour un projet peut atteindre le million. Un nombre aussi important de nœuds engendre beaucoup de transmissions inter nodales et nécessite que le puits "sink" soit équipé de beaucoup de mémoire pour stocker les informations reçues.
- **Les coûts de production :** Souvent, les réseaux de capteurs sont composés d'un très grand nombre de nœuds. Le prix d'un nœud est critique afin de pouvoir concurrencer un réseau de surveillance traditionnel. Actuellement un nœud ne coûte souvent pas beaucoup plus que 1. A titre de comparaison, un nœud Bluetooth, pourtant déjà connu pour être un système low-cost, revient environ à 10 dollar .
- **L'environnement :** Les capteurs sont souvent déployés en masse dans des endroits tels que des champs de bataille au-delà des lignes ennemies, à l'intérieur de grandes machines, au fond d'un océan, dans des champs biologiquement ou chimiquement souillés, ... Par conséquent, ils doivent pouvoir fonctionner sans surveillance dans des régions géographiques éloignées.
- **La 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,...), Redéploiement de nœuds additionnels.
- **Les contraintes matérielles :** La principale contrainte matérielle est la taille du capteur. Les autres contraintes sont que la consommation d'énergie doit être moindre pour que le réseau survive le plus longtemps possible, qu'il s'adapte aux différents environnements (fortes chaleurs, eau,...), qu'il soit autonome et très résistant vu qu'il est souvent déployé dans des environnements hostiles.
- **Les 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 normé. On utilise le plus souvent l'infrarouge (qui est license-free, robuste aux interférences, et peu onéreux), le bluetooth et les communications radioZigBee.
- **La consommation d'énergie :** Un capteur, de par sa taille, est limité en énergie ($< 1.2V$). Dans la plupart des cas le remplacement de la batterie est impossible. Ce qui veut dire que la durée de vie d'un capteur dépend grandement de la durée de vie de la batterie. Dans un réseau de capteurs (multi-sauts) chaque nœud collecte des données et envoie/transmet des valeurs. Le dysfonctionnement de quelques nœuds nécessite un changement de la topologie du réseau et un re-routage des paquets. Toutes ces opérations sont gourmandes en énergie, c'est pour cette raison que les recherches actuelles se concentrent principalement sur les

moyens de réduire cette consommation.

1.6 Points clés pour la consommation des RCSF

Comme nous venons de le mentionner, la source d'alimentation des noeuds des RCSF présente un handicap devant l'évolution de ces systèmes. A l'épuisement de la source d'alimentation, le noeud se déconnecte du réseau. En plus, dans le cas de communication multi-hop, la déconnexion d'un noeud engendre la déconnexion de tous les noeuds qui communiquent grâce à lui. Pour ce, nous avons pensé à étudier la gestion de la consommation dans ces systèmes afin de l'optimiser et par suite, augmenter la durée de vie du noeud et du réseau tout entier. Dans la littérature, il existe plusieurs méthodes qui permettent d'obtenir un système à faible consommation. Généralement elles sont classées en trois catégories suivant qu'elles touchent le matériel et/ou le logiciel. La première consiste à concevoir des composants spécifiques conçus pour consommer le minimum d'énergie : ceci est réalisé à base des techniques matérielles. Alors que la seconde méthode est basée sur des techniques logicielles qui consistent à optimiser le code afin de réduire sa consommation. Enfin la dernière méthode consiste à réaliser une synergie entre le logiciel et le matériel afin d'optimiser la consommation totale du système. Elle s'appuie sur des mécanismes matériels pour diminuer la consommation mais utilise aussi le logiciel pour réaliser une adaptation dynamique de la consommation en fonction de la charge courante du système.[2]

1.7 Classification des réseaux de capteurs sans fil

1.7.1 Selon le mode de communication

Le mode de communication utilisé dans le réseau de capteurs dépend du type d'application et des techniques utilisées pour faire acheminer l'information des capteurs à la station de base. On distingue dans cette classification trois types de communication : les réseaux de capteurs à un seul-saut (single-hop WSN), les réseaux de capteurs multi-sauts (multi-hop WSN) et les réseaux de capteurs hiérarchiques.

Dans les réseaux de capteurs à un seul-saut, les noeuds capteurs envoient les données collectées directement à la station de base sans passer par des noeuds intermédiaires. Dans ce type de réseau les noeuds envoient leurs données en utilisant une forte puissance comme illustré dans la figure 1.5 (a).

Dans les réseaux de capteurs multi-sauts, un noeud capteur envoie ses données à la station de base par l'intermédiaire de ses noeuds voisins en utilisant une petite puissance de transmission, la figure 1.5 (b) illustre un exemple sur ce type de communication. Ce type de réseau est appliqué dans plusieurs domaines d'application, mais il reste difficile à mettre en oeuvre [Shio K. S et al, 2011].

Dans les réseaux de capteurs hiérarchisés, la zone d'observation est divisée en clusters. Un clusterhead est élu pour chaque cluster. Ce dernier s'occupe de récupérer les informations auprès des capteurs dans son cluster et de les transmettre directement à la station de base figure 1.5 (c) ou via un mode multi-saut entre les clusters head figure 1.5 (d) [14].

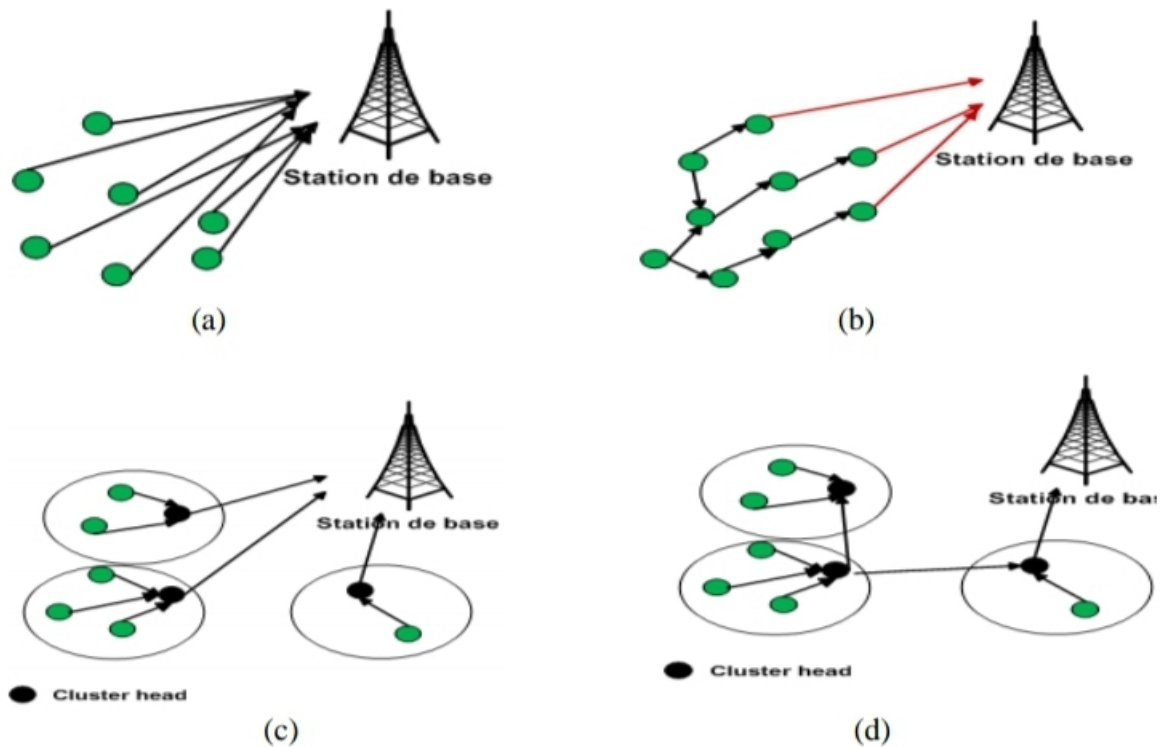


FIGURE 1.5 – les types de communication dans les réseaux de capteurs sans fil

1.8 Le routage dans le RCSF

1.8.1 Définition de routage

Le routage est un processus qui permet de sélectionner des chemins dans un réseau pour transmettre des données depuis un expéditeur jusqu'à un ou plusieurs destinataires. On parle de routage dans différents domaines : réseaux téléphoniques, réseaux électroniques (comme Internet), réseaux de transports. Le routage est le mécanisme par lequel des chemins sont sélectionnés dans un réseau pour acheminer les données d'un expéditeur jusqu'à un ou plusieurs destinataires. Le routage est une tâche exécutée dans de nombreux réseaux, tels que le réseau téléphonique, les réseaux de données électroniques comme l'Internet, et les réseaux de transports. Sa performance est importante dans les réseaux décentralisés, c'est-à-dire où l'information n'est pas distribuée par une seule source, mais échangée entre des agents indépendants. Le terme routage désigne l'ensemble des mécanismes mis en œuvre dans un réseau pour déterminer les routes qui vont acheminer les paquets d'un terminal émetteur à un terminal récepteur. On distingue généralement deux entités :

- * L'algorithme de routage.
- * Le protocole de routage[13].

1.8.2 Classification des protocoles de routage pour les RCSFs

Les protocoles de routage pour les RCSFs peuvent être classés selon plusieurs critères. La Figure 1.6 illustre une classification qui se base sur quatre critères : la structure du réseau, fonction

du protocole, l'établissement de la route, l'initiateur de la communication[16].

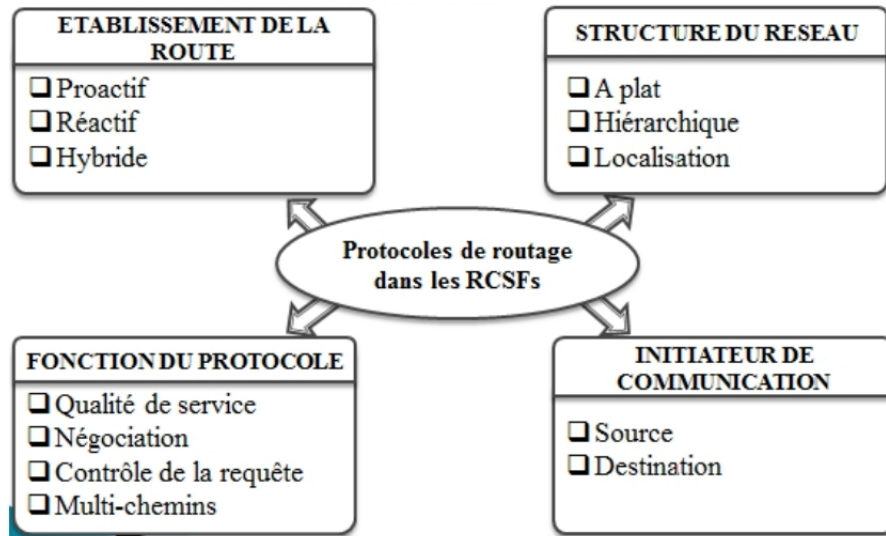


FIGURE 1.6 – Classification des protocoles de routage.

Protocoles de routage basés sur la structure du réseau

Les protocoles de routage basés sur la structure du réseau peuvent être classés en trois catégories : protocoles à plat (Flat based routing), protocoles hiérarchiques (Hierarchic based routing/Clustering based routing) et protocoles basés sur la localisation géographique (Location based routing) [16].

Routage à plat Comme illustré sur la Figure 1.7, dans cette catégorie de protocoles, les noeuds ont le même rôle et ils collaborent entre eux pour accomplir la tâche de routage. En raison du grand nombre de tels noeuds, il n'est pas faisable d'affecter un identificateur à chaque noeud.



FIGURE 1.7 – Protocoles de routage à plat.

Routage hiérarchique Comme illustré sur la Figure 1.8, dans une architecture hiérarchique, les noeuds sont regroupés en cluster et chaque cluster est orchestré par un cluster head ou chef de

cluster. C'est ce dernier qui communique les données reçues des noeuds de son cluster à la station de base. Un protocole de routage doit être conçu de telle sorte qu'il prend en considération ce découpage hiérarchique pour assurer la bonne délivrance des paquets au destinataire.

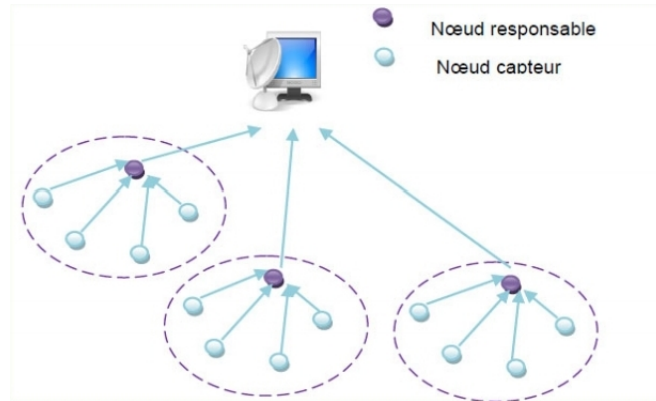


FIGURE 1.8 – Topologie hiérarchique.

Routage géographique La plupart des protocoles de routage dans les réseaux de capteurs nécessitent la localisation des noeuds capteurs. En général, ces informations sont nécessaires pour calculer la distance entre deux noeuds particuliers de sorte que la consommation d'énergie puisse être estimée. Puisque il n'y a aucun système d'adressage pour les noeuds dans les réseaux de capteurs (comme les adresses IP) et comme ils sont déployés dans une région d'une manière aléatoire, l'information de localisation de ces noeuds peut être utilisée dans le routage des données d'une manière efficace en termes d'énergie.

Classification selon l'initiateur de communication

La communication dans un réseau de capteurs peut être initiée par les noeuds sources ou par les noeuds destinataires[16].

Communication lancée par la source Dans les protocoles de communication lancée par la source, les noeuds envoient des données à la destination quand ils les ont capturées. Ces protocoles utilisent les données rapportées avec time-driven ou avec event-driven. Ceci signifie que les données sont envoyées à certains intervalles ou quand les noeuds capturent certains événements.

Communication lancée par la destination Les protocoles de communication lancée par la destination utilisent les données rapportées avec query-driven, et dans ce cas, les noeuds répondent aux requêtes envoyées par la destination ou un autre noeud différent. C'est-à-dire propager les requêtes à tous les noeuds d'une région topologique et attendre la réception des données du noeud capteur concerné dans cette région.

Classification selon l'établissement de la route

Suivant la manière de création et de maintenance des routes lors de l'acheminement des données, les protocoles de routage peuvent être séparés en trois catégories : les protocoles proactifs, les protocoles réactifs et les protocoles hybrides[16].

Protocoles proactifs Les protocoles de routage proactifs essaient de maintenir les meilleurs chemins existants vers toutes les destinations possibles (qui peuvent représenter l'ensemble de tous les noeuds du réseau) au niveau de chaque noeud du réseau.

Protocoles réactifs Les protocoles de routage réactifs (dit aussi : les protocoles de routage à la demande) créent et maintiennent des routes selon les besoins. Lorsque le réseau a besoin d'une route, une procédure de découverte de route est lancée.

Protocoles hybrides Les protocoles hybrides combinent les deux idées des protocoles proactifs et réactifs. Ils utilisent un protocole proactif pour apprendre le proche voisinage (par exemple le voisinage à deux ou à trois sauts), ainsi, ils disposent des routes immédiatement dans le voisinage. Au de là de la zone de voisinage, le protocole hybride fait appel à un protocole réactif pour chercher des routes.

1.9 Quelques définitions cruciales

A. Energie

L'énergie est considérée comme la principale et fondamentale contrainte dans les réseaux de capteurs sans fil. Chaque nœud fonctionne grâce à une batterie ayant une capacité limitée dû à sa petite taille mais également non rechargeable. L'utilisation de ces capteurs se fera, dans la plupart des cas, dans des environnements hostiles ou difficiles d'accès, ce qui ne permettra pas la récupération de celles-ci. Nous pouvons donc dire que toute utilisation de la technologie du réseau de capteur sans fil doit prendre en compte principalement de la problématique « consommation énergétique » [13].

B. Durée de vie

C'est l'intervalle de temps qui sépare l'instant de déploiement du réseau de l'instant où l'énergie du premier nœud s'épuise. Selon l'application, la durée de vie exigée pour un réseau peut varier entre quelques heures et plusieurs années [13].

D. Déploiement des nœuds

C'est un facteur dépendant de l'application qui affecte grandement les protocoles de routage. Le déploiement peut être déterministe ou aléatoire. Dans la première stratégie, les capteurs sont placés manuellement et les données peuvent donc être acheminées via des chemins prédéterminés. En revanche, avec une approche aléatoire, les capteurs sont éparpillés (lâchés d'un avion). Dans le cas d'une répartition non uniforme, une stratégie de groupement (clustering) peut s'avérer nécessaire [13].

1.10 Conclusion

Les réseaux de capteur constituent un sujet de recherche relativement récent et pour tenter de résoudre tous les problèmes soulevés dans ce contexte de nombreuses directions en termes d'algorithmes, protocoles, . . . etc. les aspects énergétiques et routage constituent les contraintes majeures.

Ce chapitre nous a permis d'avoir une vue globale sur les réseaux de capteurs sans fil notamment sur les routages et ses protocoles qui présentent un intérêt considérable vu l'évolution connue récemment dans ce domaine.

Pour répondre à notre objectif primordial qui est l'optimisation de la consommation de l'énergie dans un RCSF nous allons concevoir un algorithme optimal qui fera l'objet du chapitre qui suit.

Chapitre 2

Quelques rappels sur la théorie des graphes

2.1 introduction

La théorie des graphes est un outil puissant de modélisation et de résolution de problèmes concrets. A l'origine, la théorie des graphes était présentée comme une curiosité mathématique ; Euler lors d'une de ses promenades nocturnes a voulu tracer un itinéraire circulaire dans la ville de Königsberg. Partant d'un point donné, il voulut visiter les sept ponts de cette ville (disposés selon le schéma ci-dessous) une seule fois seulement, puis retourner à son point de départ.

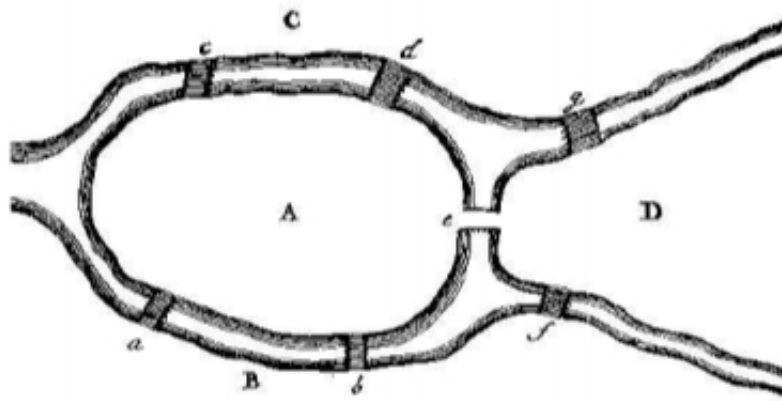


FIGURE 2.1 – les sept ponts de Königsberg

Les points A, B, C et D sont des rives. Ensuite la théorie des graphes a été utilisée pour modéliser des circuits électriques (Kirch-hoff), puis de nombreuses applications dans différents domaines tels : la chimie, la psychologie, etc [19].

2.2 Concepts de base

2.2.1 Définitions générales

Les graphes sont des concepts mathématiques utilisés pour modéliser des relations binaires entre des objets d'un même ensemble. Ils sont fréquemment utilisés pour modéliser des systèmes qui se présentent sous la forme d'un réseau. Il existe deux types de graphes : les graphes orientés et les graphes non orientés [19].

2.2.2 Graphe orienté

Un graphe orienté est un couple $(X;U)$, où X est l'ensemble des sommets du graphe et U , l'ensemble de ses arcs. X et U sont finis. L'arc est une relation entre deux sommets, dotée d'une orientation :

Si $u = (x;y)$ est un arc de U , avec $x, y \in X$, la relation est orientée de x vers y . Le graphe G est noté $G = (X;U)$ [19].

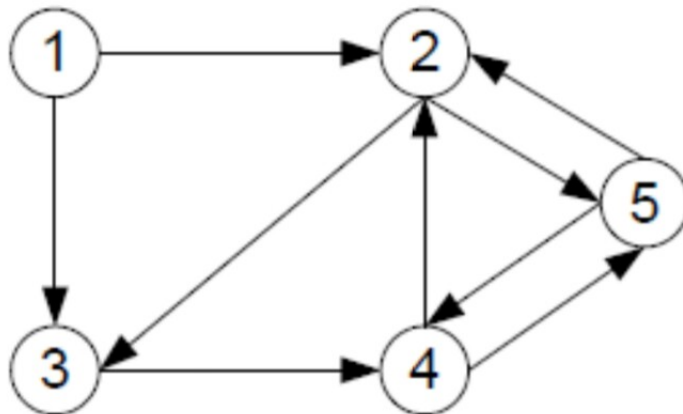


FIGURE 2.2 – Graphe orienté

2.2.3 Graphe non orienté

Un graphe non orienté est défini par le couple $(X;E)$, où X est l'ensemble des sommets du graphe et E , l'ensemble de ses arêtes .

Si x, y sont en relation, cette dernière est durite par l'arête $e=(xy)$.

Ici, le sens de la relation n'est pas invoqué.

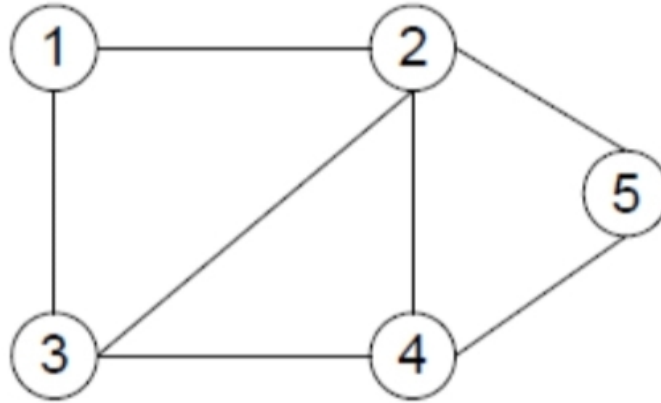


FIGURE 2.3 – Graphe non orienté

Les figures 2.2 et 2.3 représentent, respectivement, un exemple d'un graphe orienté et un exemple d'un graphe non orienté [19].

2.2.4 Arcs adjacents, arêtes adjacentes

Deux arcs (resp.arêtes) sont adjacents (resp.adjacentes) en un sommet x , si x est une extrémité commune aux deux [19].

2.3 Connexité dans les graphes

2.3.1 chaîne

Soit $G=(X, U)$ un graphe. Une chaîne joignant deux sommets x_0 et x_k dans G est une suite de sommets tels que deux sommets successifs sont reliés par une chaîne. On la note (x_0, x_1, \dots, x_k) . x_0 et x_k sont les extrémités de la chaîne commune.

On la note : $(x_0, x_1, x_2, \dots, x_k)$ et on dit que x_0 et x_k sont les extrémités de la chaîne[19].

2.3.2 Chemin

C'est la suite de sommets et d'arcs $(x_i, u_i, x_{i+1}, \dots, x_k, u_k, x_{k+1}, \dots, x_j)$ tels que $x_k = I(u_k)$ et $x_{k+1} = T(u_k)$ [19].

2.3.3 Cycle

Un cycle est une chaîne simple dont les extrémités coïncident. On le note $(x_0, x_1, x_2, \dots, x_k = x_0)$ [19].

2.3.4 Circuit

Un circuit est un chemin dont les extrémités sont confondues. On le note par $(x_0, x_1, x_2, \dots, x_k = x_0)$ [19].

2.3.5 connexité

Un graphe est connexe si $\forall x, y \in X$, il existe une chaîne entre x et y [19].

2.3.6 Forte connexité

On définit la forte connexité dans un graphe par une relation entre deux sommets de la manière suivante :

Deux sommets x et y ont une relation de forte connexité

$$\Leftrightarrow$$

Il existe un chemin de x à y et un chemin de y à x ; ou bien $x=y$ [19].

2.3.7 Arbres et forêts

Un **arbre** est un graphe connexe et acyclique (c'est-à-dire un graphe sans cycle) , Ainsi un arbre est nécessairement simple, une chaîne élémentaire est en particulier un arbre. Les arbres ont des propriétés, on distingue toujours par n_G et m_G où n le nombre de sommets d'un arbre et m son nombre d'arêtes [15].

Caractérisation des arbres

soit $G = (V; E)$ un arbre, alors :

1. G est acyclique et on a $|V| - 1$ arêtes
2. G est un graphe connexe minimal (chaque arête est un isthme [un isthme est une arête dont la suppression rend le graphe non connexe])
3. G est un graphe maximal sans cycles (l'addition d'une arête quelconque crée un cycle),
4. Toute paire de sommets de G est connectée par une chaîne unique [15].

Arbres couvrants d'un graphe

un arbre couvrant d'un graphe $G = (V; E)$ est un graphe $A(V'; E')$ partiel de G qui est un arbre et dont $V' = V$ [15].

2.4 Quelques types de graphes

2.4.1 Graphe valué

graphe où des réels sont associés aux arêtes. Dans cet exposé, on ne considérera que des valuations positives.

2.4.2 Graphe Complet

Un graphe G est dit complet si tous les sommets sont deux à deux adjacents c'est-à-dire tous les sommets du graphe sont relié entre eux comme le montre la figure 2.4 [20].

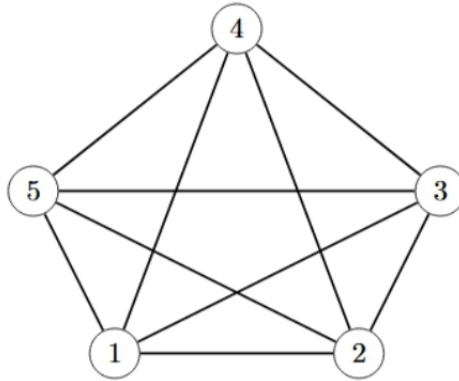


FIGURE 2.4 – Graphe complet

2.4.3 Sous-graphe

Un sous graphe est une partie d'un graphe, Un sous-graphe $H(U, F)$ de $G(V, E)$ est un graphe tel que :

- U est un sous-ensemble V ($U \subseteq V$)[20].

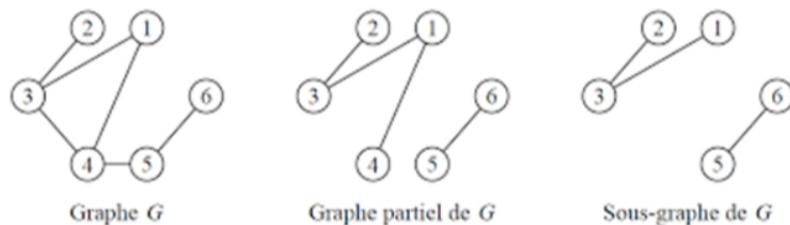


FIGURE 2.5 – Sous-graphe, graphe partiel

2.4.4 Sous-graphe partiel

Il s'agit d'un sous graphe. C'est concrètement un graphe au quel on a enlevé des sommets et leurs arêtes incidentes (figure 1.7). Un sous-graphe partiel de $G = (V; E)$ est un graphe partiel d'un sous-graphe [20].

2.4.5 graphe planaire

Un graphe est planaire s'il peut être tracé dans un plan sans qu'aucun croisement entre les arêtes [20].

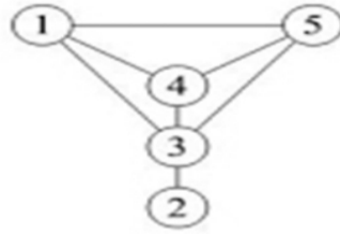


FIGURE 2.6 – Graphe planaire

2.4.6 Graphe biparti

Un graphe $G = (V; E)$ est dit biparti si l'ensemble de ses sommets V peut être partitionner en deux sous-ensembles V_1 et V_2 , de sorte que les sommets de même sous ensemble ne sont pas adjacents [20].

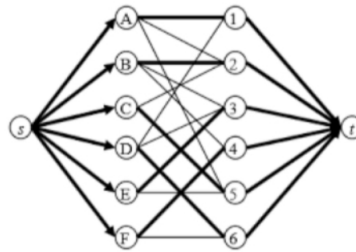


FIGURE 2.7 – Graphe biparti

2.5 Plus courts chemins

2.5.1 Problème du plus court chemin

On se place dans le cas des graphes orientés valués $G = (S; A; V)$. Mais les résultats et algorithmes présentés se généralisent facilement aux cas des graphes non orientés valués. Une autre solution consiste à transformer le graphe non-orienté en un graphe orienté en remplaçant une arête entre deux sommets par deux arcs de sens inverse entre ces sommets[18].

Définition

-Le coût ou poids d'un chemin $c = \langle s_0; s_1; \dots; s_k \rangle$ est égale à la somme des valuations des arcs composant le chemin, c'est à dire,

$$L(c) = \sum_{i=1}^k V(s_{i-1}, s_i)$$

-Le coût d'un plus court chemin entre deux sommets s_i et s_j est noté $\delta(s_i; s_j)$ et est défini par :

$$\delta(s_i; s_j) = \begin{cases} \min\{L(c)/c = \text{chemin de } s_i \text{ à } s_j\}, & \text{s'il existe au moins un chemin entre } s_i \text{ et } s_j \\ +\infty, & \text{sinon} \end{cases} \quad (2.1)$$

Dans la recherche d'un plus court chemin de x à y, trois cas peuvent se présenter :

- Il n'existe aucun chemin de x à y (par exemple, si x et y appartiennent à deux composantes fortement connexes différentes de G).
- Il existe des chemins de x à y mais pas de plus court chemin.
- Il existe un plus court chemin de x à y.

2.6 Algorithme de recherches de Plus courts chemins

Algorithme de Dijkstra

(Recherche d'un plus court chemin de la racine s à tous les autres sommets i dans un réseau ayant toute les évaluations positive)[21].

Données : $R=(S,A,V)$, s :Racine

Résultats : π : Les plus courtes distances, B : Arborescence, C : Sous-ensemble de sommets accessibles a partir de s.

a) Initialisation : $C=\{s\}, \pi(s)=0, B= \pi(j)=\{ v_{sj} \text{ si } (s,j) \in A, +\infty \text{ si non.}$

b) Procédure de calcul :

(.) chercher un sommet $i \in C^C$ vérifiant : $\pi(i)=\min\{ \pi(j) \} j \in C^C$

(..)Poser : $C=C \cup \{i\}$ et $\pi(j) = \min\{\pi(j), \pi(i) + v_{ij}\} j \in \tau_i \cap C^C$

C) Test d'arrêt : si $|C|=|S|$, Terminer, le Réseau R contient au moins un plus court chemin de s à tout sommet i.

Sinon : retourner en l'étape(b).

2.7 Conclusion

Dans ce chapitre nous avons défini les différentes notions et définitions relatives à la théorie des graphes ainsi, nous avons proposé quelques algorithmes de base que nous allons utiliser dans le chapitre suivant.

Chapitre 3

Protocole de routage basée sur d'algorithme de Dijkstra dans un RCSFs

3.1 introduction

En général, le routage consiste à trouver un chemin entre une source de données et le destinataire de ces données. Dans les réseaux de capteurs, un protocole de routage doit assurer l'efficacité énergétique, pour cela il faut maintenir les routes sans dépenser trop d'énergie et il doit tenir compte du type de communications induite par l'application. Outre le fait que la quantité de données échangées est très faible par rapport aux applications de types réseaux ad hoc, notons que le trafic est particulièrement prévisible puis qu'il va des nœuds vers la station de base ou du la station de base vers les nœuds.

Dans ce chapitre, nous allons Implémenté l'algorithme de Dijkstra dans les RCSFs pour déterminer un plus court chemin de s à tous les autres sommets i [16].

3.2 Modèle du réseau

Le réseau considéré est un ensemble de nœuds déployées aléatoirement dans une zone de capture, ou les nœuds capteurs envoient ses données à la station de station de base par par un intermédiaire de ses nœuds voisin.

3.3 Solution proposée

la modélisation de notre problème qui est la minimisation d'énergie dans les RCSFs, nous à conduite à l'étude de sommets et d'arcs, sachant que les noeuds capteurs sont dotés d'une petite batterie.

Dans ce fait, nous avons utilisé l'un des algorithmes de théorie des graphes qui est l'algorithme de dijkstra, qui nous permettra de détermine les routes à emprunter avec un chemin de plus faible poids.

3.4 Hypothèses

Notre solution se base sur les hypothèses suivantes :

- Tous les noeuds ont le même niveau énergétique.
- Les noeuds déployées aléatoirement dans une zone de capture.
- Le graphe valués connexe ne contient pas de circuit absorbant.
- Chaque noeud peut atteindre la station de base avec communication multi-sauts.
- Les capteurs sont fixe.
- Chaque noeud possède un identifiant unique.
- La mort de chaque capteur n'est causée que par l'épuisement de son énergie.
- la station de base est vue comme une ressource non limitée ni épuisable(l'unité de stockage , l'énergie).

3.5 Détail de la solution proposée

3.5.1 Objectif de notre modélisation

Modélisation de système par un graphe.

Pour cela nous avons modéliser notre problème avec l'algorithme de dijkstra qui détermine la distance minimale et permettre d'acheminer notre information d'un noeud capteur vers une station de base avec peut d'énergie.

Soit un graphe valué $G=(S,A,V)$; un graphe représentant une relation entre les capteurs du réseau.

- S :ensemble fini dont les élément sont appelées des sommets.
- A :ensemble fini dont les élément sont appelées des arcs.
- V :la distance entre deux sommets,
- . ou on associe a chaque arc $A=(i,j)$ une longueur π_{ij} .
- . C :sous ensemble de sommets accessibles a partir de s.

Dans le cadre de notre modélisation, nous allons considérer qu'on a une seule station de base et plusieurs noeuds de capteur.

Un exemple d'application de cet algorithme sur un graphe

La découverte d'un plus court chemin d'un noeud source à une station de base, se fait à l'aide de l'algorithme de dijkstra jusqu'à ce que tout les sommets soient parcus.

pour le graphe si dessous on suppose que on a sept noeud de capteur $C=\{s_0,s_1,s_2,s_3,s_4,s_5,s_6\}$ et une seul station de base qui est le noeud s_7 .

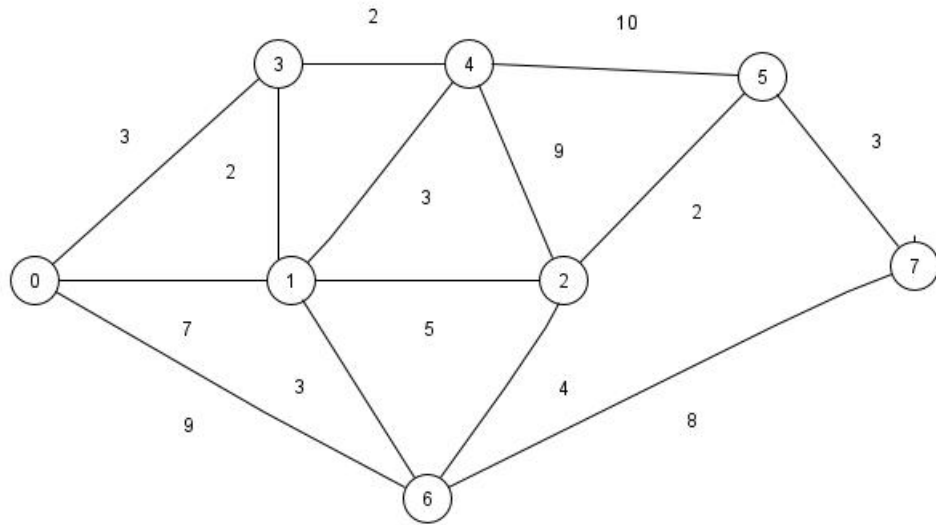


FIGURE 3.1 – Réseau associé après la modélisation.

Résolution on Applique l'algorithme de dijkstra pour déterminer un plus court chemin d'un sommet s_0 à tous les autres sommets j en se basent sur le noeud s_7 .

Initialisation

On pose $C = \{ s_0 \}$; $\pi(s_0) = 0$; $\pi(x) = +\infty \forall x \notin C$.

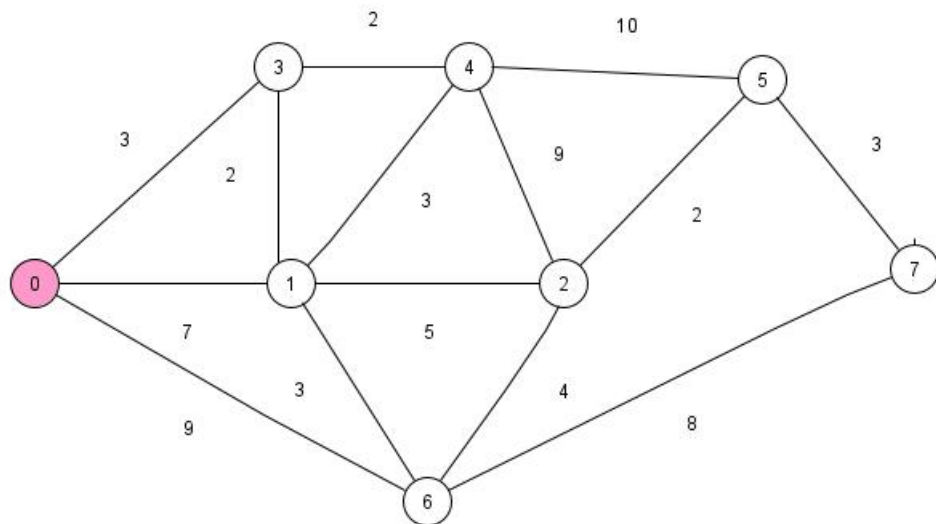


FIGURE 3.2 – Réseau obtenu après l'initialisation.

. Itération 1 :

On examine les sommets voisins de sommet s_0

$$\pi(s_3)=0+3=3; \pi(s_1)=0+7=7; \pi(s_6)=0+9=9$$

$$\min\{ \pi(s_3); \pi(s_1); \pi(s_6) \} = \pi(s_3) = 3.$$

$$C=\{s_0, s_3\}$$

$$\pi(s_3) = 3.$$

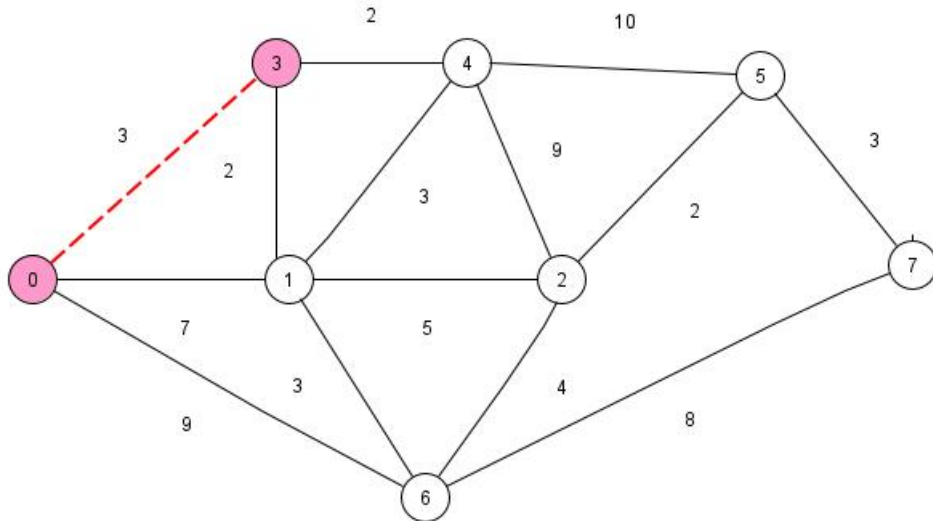


FIGURE 3.3 – Réseau obtenu à la première itération.

. Itération 2 :

On examine les sommets voisins de sommet s_0 et s_3

$$\pi(s_4)=3+2=5; \pi(s_1)=3+2=5; \pi(s_6)=0+9=9$$

$$\min\{ \pi(s_4); \pi(s_1); \pi(s_6) \} = \pi(s_1)=5.$$

$$C=\{ s_0, s_3, s_1 \}$$

$$\pi(s_1)=5.$$

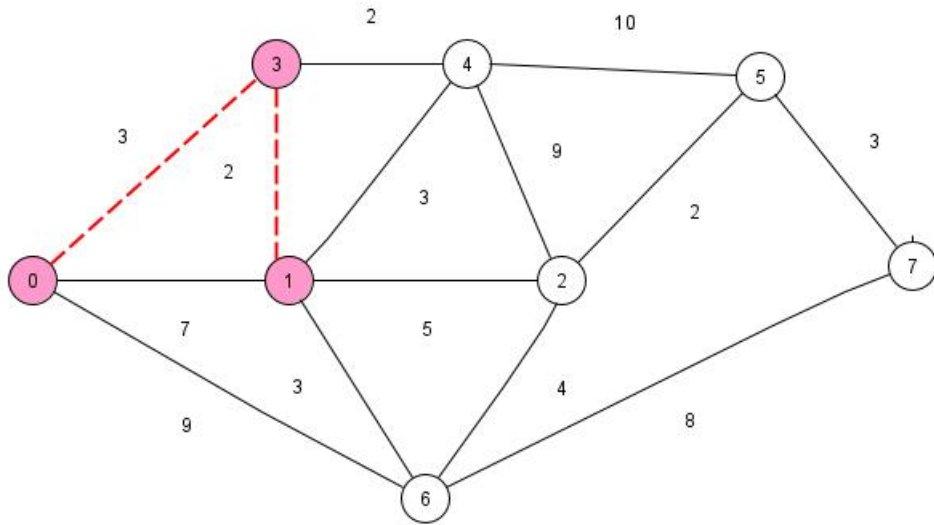


FIGURE 3.4 – Réseau obtenu à la deuxième itération.

. Itération 3 :

On examine les sommets voisins de sommet s_0 et s_3 et s_1

$$\pi(s_4)=3+2=5; \pi(s_2)=5+5=10; \pi(s_6)=5+3=8$$

$$\min\{ \pi(s_4); \pi(s_2); \pi(s_6) \} = \pi(s_4)=5.$$

$$C=\{ s_0, s_3, s_1, s_4 \}$$

$$\pi(s_4)=5.$$

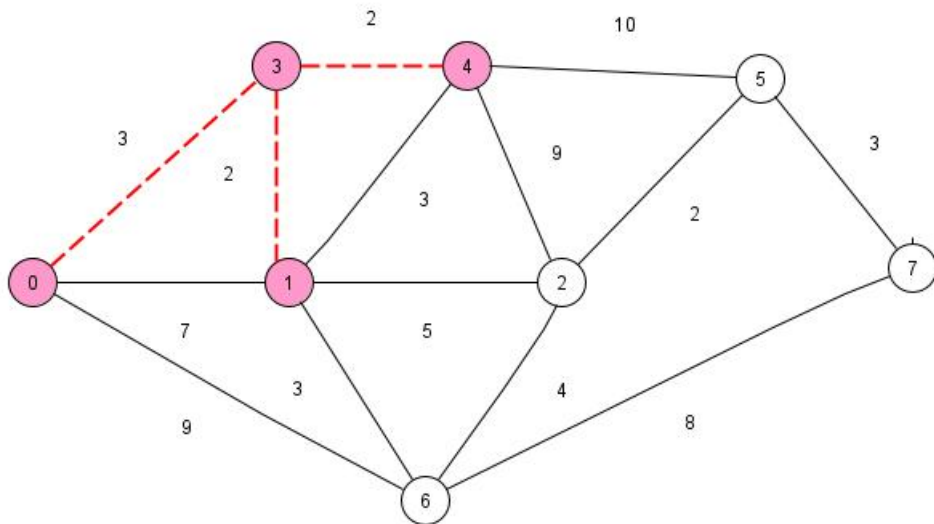


FIGURE 3.5 – Réseau obtenu à la troisième itération

. Itération 4 :

On examine les sommets voisins de sommet s_0 et s_3 et s_1 et s_4

$$\pi(s_5)=5+10=15; \pi(s_2)=5+5=10; \pi(s_6)=5+3=8$$

$$\min\{ \pi(s_5); \pi(s_2); \pi(s_6) \} = \pi(s_6)=8.$$

$$C=\{ s_0, s_3, s_1, s_4, s_6 \}$$

$$\pi(s_6)=8.$$

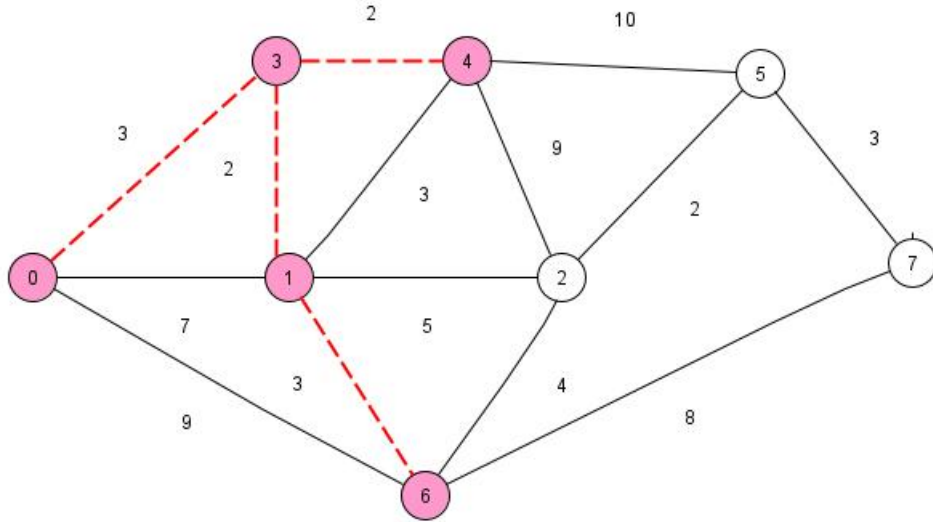


FIGURE 3.6 – Réseau obtenu à la quatrième itération.

. Itération 5 :

On examine les sommets voisins de sommet s_0 et s_3 et s_1 et s_4 et s_6

$$\pi(s_5)=5+10=15; \pi(s_2)=5+5=10; \pi(s_6)=5+3=8$$

$$\min\{ \pi(s_5); \pi(s_2); \pi(s_6) \} = \pi(s_2)=10.$$

$$C=\{ s_0, s_3, s_1, s_4, s_6, s_2 \}$$

$$\pi(s_2)=10.$$

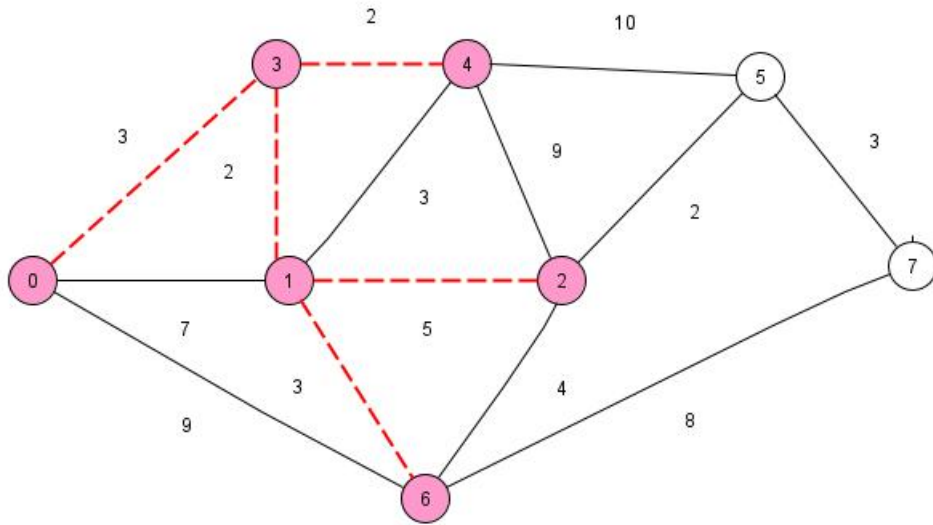


FIGURE 3.7 – Réseau obtenu à la cinquième itération.

. Itération 6 :

On examine les sommets voisins de sommet s_0 et s_3 et s_1 et s_4 et s_6 et s_2

$$\pi(s_5)=10+2=12; \pi(s_7)=8+8=16$$

$$\min\{ \pi(s_5); \pi(s_7) \} = \pi(s_5)=12.$$

$$C=\{ s_0, s_3, s_1, s_4, s_6, s_2, s_5 \}$$

$$\pi(s_5)=10.$$

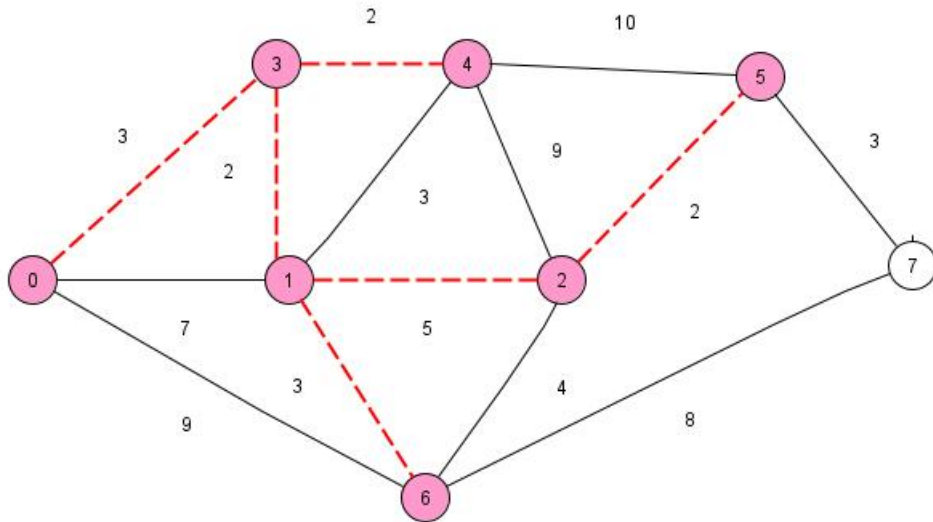


FIGURE 3.8 – Réseau obtenu à la sixième itération.

. Itération 7 :

On examine les sommets voisins de sommet s_0 et s_3 et s_1 et s_4 et s_6 et s_2 et s_5

$$\pi(s_7)=10+2=15$$

$$C=\{ s_0, s_3, s_1, s_4, s_6, s_2, s_5, s_7 \}$$

tout les sommet sont marquer, alors on s'arrête, les plus courts chemin sont tels qu'il illustre ce dernier graphe

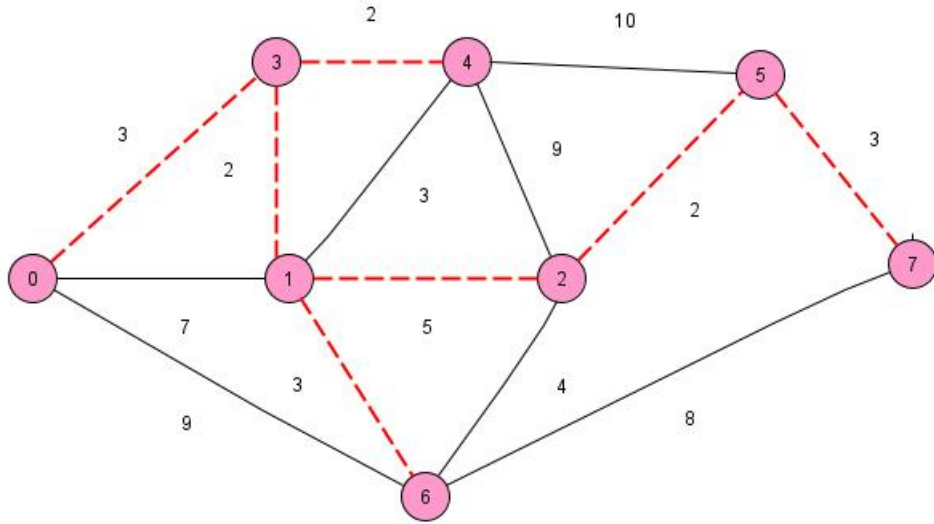


FIGURE 3.9 – Réseau obtenu à la septième itération.

on résume les itérations de déroulement de l'algorithme de dijkstra pour déterminer un plus court chemin d'un sommet s_0 à tous les autres sommets j en se basant sur le noeud s_7 .

s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	<i>choix</i>
0	—	—	—	—	—	—	—	0(0)
	$7V_{s_0}$	—	$3V_{s_0}$	—	—	$9V_{s_0}$	—	3(3)
	$5V_{s_3}$	—		$5V_{s_3}$	—	$9V_{s_0}$	—	1(5)
		$10V_{s_1}$		$5V_{s_3}$	—	$8V_{s_1}$	—	4(5)
		$10V_{s_1}$			$15V_{s_4}$	$8V_{s_1}$	—	6(8)
		$10V_{s_1}$			$15V_{s_4}$		$16V_{s_6}$	2(10)
					$12V_{s_2}$		$16V_{s_6}$	5(12)
							$15V_{s_5}$	7(15)

(3.1)

3.6 Application sur le C++

On va programmer l'algorithme de dijkstra pour obtenir un plus court chemin d'un noeud capteur à une station de base d'un réseau donné, et pour le faire nous utiliserons le langage C++.

3.6.1 Présentation de langage C++

Apparu au début des années 90, le langage C++ est actuellement l'un des plus utilisés dans le monde, aussi bien pour les applications scientifiques que pour le développement des logiciels. En tant qu'héritier du langage C, le C++ est d'une grande efficacité. Mais il a en plus des fonctionnalités puissantes, comme par exemple la notion de classe, qui permet d'appliquer les techniques de la programmation-objet[22].

3.6.2 Application numérique

Nous finissons par restituer les différents résultats de recherche du plus court chemin sur le réseau, obtenus à travers l'exécution de l'algorithme de dijkstra implémenté sous le langage C++.

CHAPITRE 3. PROTOCOLE DE ROUTAGE BASÉE SUR D'ALGORITHME DE DIJKSTRA DANS UN RCSFS

On introduit la capacité de chaque arc jusqu'à atteindre le dernier.

```
C:\Users\pc\Desktop\dijkstra\bin\Debug\dijkstra.exe
Nombre de noeuds du graphe ?8
Saisie de la matrice des distance
Saisie les distance entre les noeuds
Saisie 9999 en cas d'absence direct entre 2 noeuds differents
Saisie la valeur 0 pour la distance d'un a meme noeud
distance arc (0,0)
0
distance arc (0,1)
9999
distance arc (0,2)
9999
distance arc (0,3)
3
distance arc (0,4)
9999
distance arc (0,5)
9999
distance arc (0,6)
9
distance arc (0,7)
9999
distance arc (1,0)
7
distance arc (1,1)
0
distance arc (1,2)
5
distance arc (1,3)
2
distance arc (1,4)
3
distance arc (1,5)
9999
distance arc (1,6)
3
distance arc (1,7)
9999
distance arc (2,0)
9999
```

FIGURE 3.10 – Introduire les données.

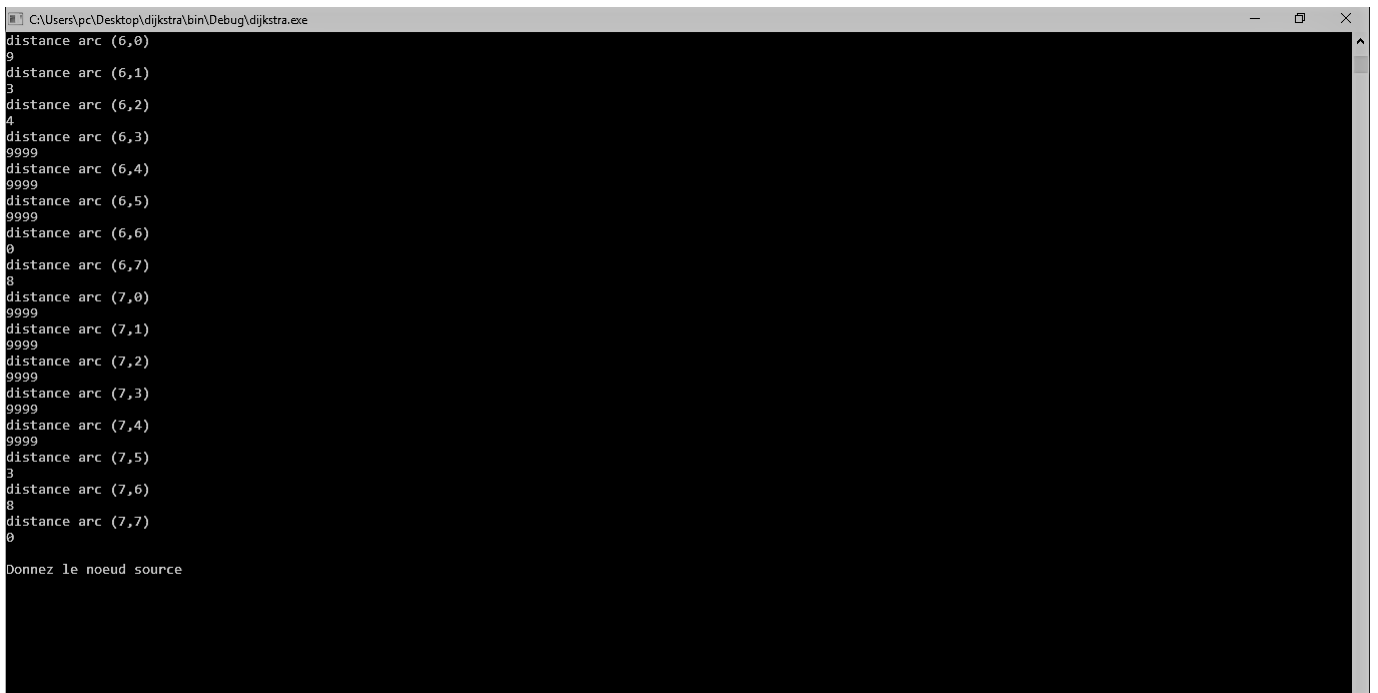
CHAPITRE 3. PROTOCOLE DE ROUTAGE BASÉE SUR D'ALGORITHME DE DIJKSTRA DANS UN RCSFS

```
C:\Users\pc\Desktop\dijkstra\bin\Debug\dijkstra.exe
distance arc (2,0)
9999
distance arc (2,1)
5
distance arc (2,2)
0
distance arc (2,3)
9999
distance arc (2,4)
9
distance arc (2,5)
2
distance arc (2,6)
4
distance arc (2,7)
9999
distance arc (3,0)
3
distance arc (3,1)
2
distance arc (3,2)
9999
distance arc (3,3)
0
distance arc (3,4)
2
distance arc (3,5)
9999
distance arc (3,6)
9999
distance arc (3,7)
9999
distance arc (4,0)
9999
distance arc (4,1)
```

FIGURE 3.11 – Introduire les données.

```
C:\Users\pc\Desktop\dijkstra\bin\Debug\dijkstra.exe
distance arc (4,0)
9999
distance arc (4,1)
3
distance arc (4,2)
9
distance arc (4,3)
2
distance arc (4,4)
0
distance arc (4,5)
10
distance arc (4,6)
9999
distance arc (4,7)
9999
distance arc (5,0)
9999
distance arc (5,1)
9999
distance arc (5,2)
2
distance arc (5,3)
9999
distance arc (5,4)
10
distance arc (5,5)
0
distance arc (5,6)
9999
distance arc (5,7)
3
distance arc (6,0)
9
distance arc (6,1)
3
distance arc (6,2)
```

FIGURE 3.12 – Introduire les données.



```
C:\Users\pc\Desktop\dijkstra\bin\Debug\dijkstra.exe
distance arc (6,0)
9
distance arc (6,1)
3
distance arc (6,2)
4
distance arc (6,3)
9999
distance arc (6,4)
9999
distance arc (6,5)
9999
distance arc (6,6)
0
distance arc (6,7)
8
distance arc (7,0)
9999
distance arc (7,1)
9999
distance arc (7,2)
9999
distance arc (7,3)
9999
distance arc (7,4)
9999
distance arc (7,5)
3
distance arc (7,6)
8
distance arc (7,7)
0
Donnez le noeud source
```

FIGURE 3.13 – Introduire les données.

Résultat

On donne le noeud source est : s_0
le programme nous donne la distance minimale de s_0 a tous les noeud de graphe en particulier le noeud s_7 qui est la station de base.

CHAPITRE 3. PROTOCOLE DE ROUTAGE BASÉE SUR D'ALGORITHME DE DIJKSTRA DANS UN RCSFS

```
C:\Users\pc\Desktop\dijkstra\bin\Debug\dijkstra.exe
distance arc (6,7)
8
distance arc (7,0)
9999
distance arc (7,1)
9999
distance arc (7,2)
9999
distance arc (7,3)
9999
distance arc (7,4)
9999
distance arc (7,5)
3
distance arc (7,6)
8
distance arc (7,7)
0

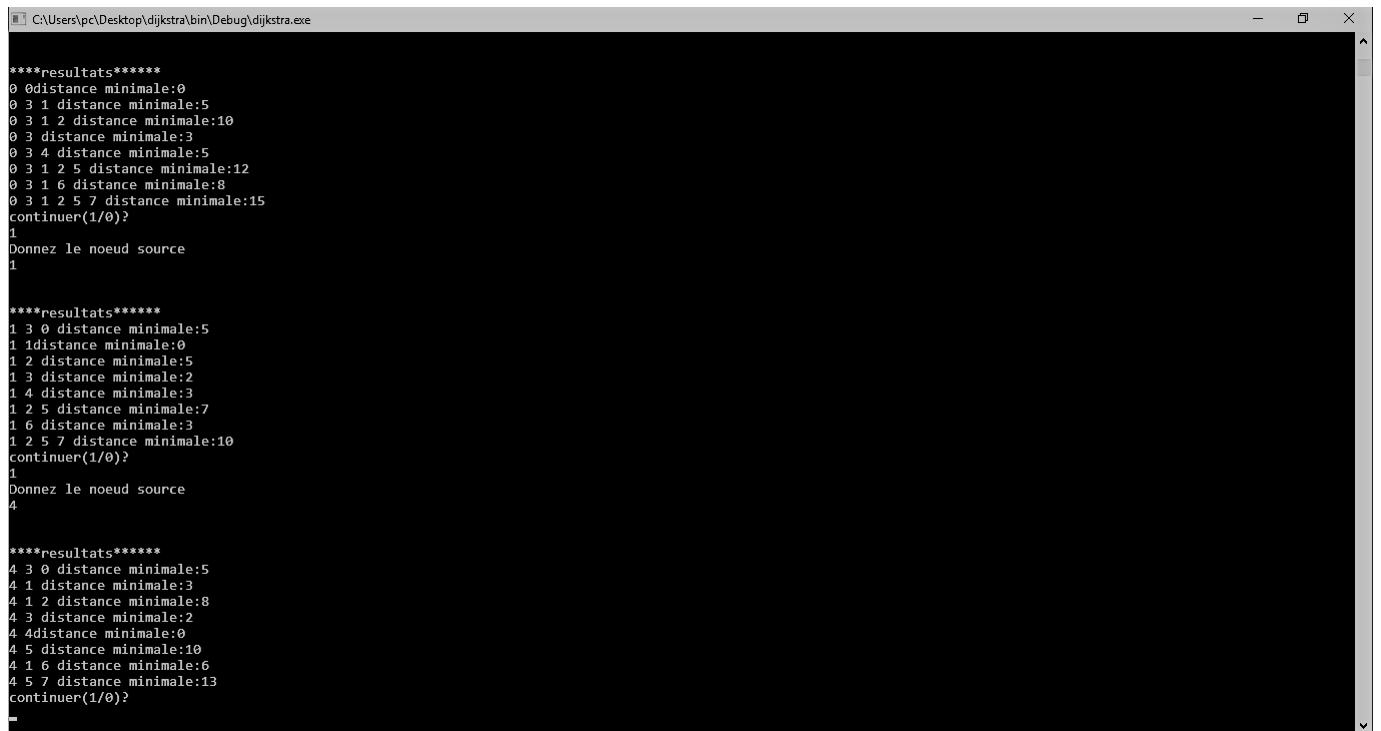
Donnez le noeud source
0

****resultats****
0 0 distance minimale:0
0 3 1 distance minimale:5
0 3 1 2 distance minimale:10
0 3 distance minimale:3
0 3 4 distance minimale:5
0 3 1 2 5 distance minimale:12
0 3 1 6 distance minimale:8
0 3 1 2 5 7 distance minimale:15
continuer(1/0)?
```

FIGURE 3.14 – Résultat.

on tape 1 pour continue
on donne un autre noeud source s_1

CHAPITRE 3. PROTOCOLE DE ROUTAGE BASÉE SUR D'ALGORITHME DE DIJKSTRA DANS UN RCSFS



```
C:\Users\pc\Desktop\dijkstra\bin\Debug\dijkstra.exe

***resultats***
0 0 distance minimale:0
0 3 1 distance minimale:5
0 3 1 2 distance minimale:10
0 3 distance minimale:3
0 3 4 distance minimale:5
0 3 1 2 5 distance minimale:12
0 3 1 6 distance minimale:8
0 3 1 2 5 7 distance minimale:15
continuer(1/0)?
1
Donnez le noeud source
1

***resultats***
1 3 0 distance minimale:5
1 1 distance minimale:0
1 2 distance minimale:5
1 3 distance minimale:2
1 4 distance minimale:3
1 2 5 distance minimale:7
1 6 distance minimale:3
1 2 5 7 distance minimale:10
continuer(1/0)?
1
Donnez le noeud source
4

***resultats***
4 3 0 distance minimale:5
4 1 distance minimale:3
4 1 2 distance minimale:8
4 3 distance minimale:2
4 4 distance minimale:0
4 5 distance minimale:10
4 1 6 distance minimale:6
4 5 7 distance minimale:13
continuer(1/0)?
0
```

FIGURE 3.15 – Résultat.

on tape 1 pour continue
on donne un autre noeud source s_4
si on veut que le processus s'arrête on tape 0.


```
C:\Users\pc\Desktop\dijkstra\bin\Debug\dijkstra.exe
0 3 1 distance minimale:5
0 3 1 2 distance minimale:10
0 3 distance minimale:3
0 3 4 distance minimale:5
0 3 1 2 5 distance minimale:12
0 3 1 6 distance minimale:8
0 3 1 2 5 7 distance minimale:15
continuer(1/0)?
1
Donnez le noeud source
1
****resultats*****
1 3 0 distance minimale:5
1 1 distance minimale:0
1 2 distance minimale:5
1 3 distance minimale:2
1 4 distance minimale:3
1 2 5 distance minimale:7
1 6 distance minimale:3
1 2 5 7 distance minimale:10
continuer(1/0)?
1
Donnez le noeud source
4
****resultats*****
4 3 0 distance minimale:5
4 1 distance minimale:3
4 1 2 distance minimale:8
4 3 distance minimale:2
4 4 distance minimale:0
4 5 distance minimale:10
4 1 6 distance minimale:6
4 5 7 distance minimale:13
continuer(1/0)?
0
Process returned 0 (0x0)   execution time : 1214.485 s
Press any key to continue.
```

FIGURE 3.16 – Résultat.

3.7 Conclusion

Ce dernier chapitre est consacré à la modélisation d'un réseau de capteur sans fil par un graphe, sachant que plus le noeud source s'éloigne du noeud puit, plus de consommation d'énergie, et dans le contexte de l'étude de l'énergie et de la façon de maintenir la durée de vie du réseau, on propose une programmation de l'algorithme de Dijkstra sur le langage C++ qui calcule le meilleur chemin ayant un poids minimal, entre la source et la destination comme une solution.

3.8 Conclusion générale

Dans ce travail, nous sommes intéressé à la minimisation de l'énergie dans un réseau de capteur sans fil, dans ce derniers les noeuds capteurs sont alimentés par des batteries à faible capacité irremplaçable car ils déploient dans des zones inaccessible.

Dans un premier lieu, 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. A cette fin, le routage est considéré comme l'un des aspects les plus importants à étudier. Dans le chapitre suivant, nous allons mettre l'action sur le routage dans les réseaux de capteurs sans fil.

Notre bute est de prolonger la durée de vie du réseau de capteurs en minimisant la consommation d'énergie, sachant que plus le noeud source s'éloigne du noeud puit, plus la consommation de l'énergie et dans ce contexte De nombreux algorithmes, et protocoles ont été proposés dans la littérature pour traiter les problématiques de la minimisation de l'énergie dissipée par les capteurs on a implémenter un protocole de routage basée sur l'algorithme de djikstra pour trouver le meilleur chemin ayant un poids minimal entre la source et la destination.

Bibliographie

- [1] KECHAR Bou Abdellah, Problématique de la consommation d'énergie dans les réseaux de capteur sans fil, mémoire fin d'étude université d'Oran, 2010
- [2] KHAOULA Lahmar, M F MEGDICHE, C. BELLOUDY, M. ABID, M AUGUIN, Estimation de la consommation dans les réseaux de capteurs sans fils : étude de cas, article.
- [3] conception d'un RCSF, mémoire pour le projet d'étude à université Mohammed V école normale supérieure d'enseignement technique, RABAT 2015.
- [4] Jean Mickaël Lebreton, Systèmes et protocoles de télé-réveil appliqués à l'optimisation énergétique des réseaux de capteurs sans fil, thèse de doctorat, université de réunion, 2017.
- [5] Youcef Touati, A. ALI cherif, et B. Daachi, « gestion énergétique dans les réseaux de capteurs sans fil » livre de collection réseaux et Télécommunications, 2017.
- [6] KHALILI Zeyneb, BOUCHRA Meryem « Une technique d'optimisation de la consommation d'énergie dans les réseaux de capteurs sans fil », mémoire fin d'étude université ahmed-Draia, Adrar, 2019.
- [7] « gestion de la mobilité dans les réseaux de capteur sans fils », thèse de doctorat Lyon 1
- [8] Cheicktidjanekone, « conception de l'architecture d'un réseau de capteur sans fil de grande dimension », thèse.
- [9] Mehdi Bouallegue. « Protocoles de communication et optimisation de l'énergie dans les réseaux de capteurs sans fil » thèse de doctorat, université Bretagne Loire 2016
- [10] « Routage à basse consommation énergétique dans les réseaux de capteur sans fil à base de l'algorithme de dijkstra », mémoire fin d'étude à l'université de Bejaia, 2014.
- [11] YACINE CHALLAL, [http ://creativecommons.org/licenses/by-nc/2.0/fr/](http://creativecommons.org/licenses/by-nc/2.0/fr/), site.
- [12] « Optimisation d'accès au médium afin d'économiser de l'énergie dans les réseaux de capteurs sans fils ».
- [13] « Techniques de conservation d'énergie pour les réseaux de capteur sans fil » mémoire fin d'étude université de Saida.
- [14] Thèse intitulée « Prolongation de la Durée de Vie des Batteries dans les Réseaux de Capteurs Sans Fil (RCSF) » université badji mokhtar-annaba année 2015.
- [15] Mémoire de Fin d'études Master Filière : Informatique Thème : « Reconnaissance des motifs dans des graphes EMF » Juillet 2019 Université de 8 Mai 1945 – Guelma -
- [16] Thème « Optimisation de la consommation d'énergie dans les réseaux de capteurs sans fil » mémoire fin cycle à l'université de Bejaia, 2014.
- [17] Thèse l'Université Henri Poincaré, Nancy I « Conception de l'architecture d'un réseau de capteurs sans fil de grande dimension » année 2011.

- [18] IUT Lyon Informatique « Théorie des Graphes » 2011-2012.
- [19] MÉMOIRE DE MASTER 2 en MATHÉMATIQUES Thème : « Algorithmes linéaires du nombre de broadcast domination de quelques classes de graphes » UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU 2014/2015.
- [20] MÉMOIRE DE MASTER Thème : « Appariement de Graphes Pondérés Approche Spectrale » UNIVERSITÉ Dr. TAHAR MOULAY SAIDA, 2019
- [21] cour M.Taaouinet 2019/2020.
- [22] Cours d'informatique : «C++ : LES BASES» O. Marguin — 2003/2004.