

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE SAAD DAHLAB DE BLIDA
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE



MEMOIRE DE FIN D'ETUDES

Pour l'obtention

D'un Diplôme de Master en Informatique

Option : Ingénierie Logiciel

THÈME :

**Apprentissage par renforcement pour
l'optimisation de la consommation d'énergie
dans un contexte IoT**

Réalisé par :

M^{elle} .ELY BOUCELHAB Khadijetou

M^{elle} .OULD BABAALI Rania

Encadré par :

Mr. KAMECHE Abdallah

2018/2019

Dédicaces

Dédicaces

À mon Dieu :

À l'éternel, le tout puissant de m'avoir aidé à arriver au bout de mes études, lui qui m'a accompagné dès le début jusqu'à la fin, il est mon ombre à ma main droite.

À mon père :

Mon père ELY MOHAMED, qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Dieu faire en sorte que ce travail porte son fruit ; Merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.

À ma mère :

Ma mère FATIMETOU SADVA, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mon frère Sidi et Mes sœurs : Ezzi ♥Meimouna ♥Sedigha ♥Adita qui n'ont cessé d'être pour moi des exemples de persévérance, de courage et de générosité.

À toute ma famille : mes grands-parents, mes tantes, mes oncles, mes cousins et mes cousines.

Mes Ami(e)s

À tous les étudiants de la promotion 2018/2019 IL.

Et à tous ceux qui ont contribué de près ou de loin pour que ce projet soit possible, je Vous dis merci.

Enfin Toute personne qui m'aime et que j'aime ...

KHadijetou ♥

Dédicaces

Je dédie ce travail à ma chère maman pour avoir toujours été un exemple, pour les sacrifices qu'elle a faits et pour le soutien qu'elle m'a apporté, à mon père pour avoir cru en moi, et à toute ma famille qui a toujours été là pour moi.

A mes chères tantes, Aicha qui est une deuxième maman pour moi et Baya que j'apprécie et je remercie infiniment pour tout ce qu'elle a fait pour moi.

A mes amis Sihem, Mouna et Abdou qui ont eu un impact énorme sur mon parcours scolaire.

A mes deux meilleures amies Iméne et Maroua qui sont des sœurs pour moi, je suis si chanceuse de les avoir

A Manid qui m'a soutenu tout au long de cette année, et avec qui j'aimerai partager cette joie.

A mon binôme khadijatou d'avoir fait parti de ce travail.

Rania

Remerciement

REMERCIEMENT

*Nous remercions en premier lieu le bon Dieu le tout puissant et miséricordieux,
Qui nous a donné la force, la patience et le courage d'accomplir ce modeste
travail.*

Nous voudrions présenter nos remerciements vivement à :

*Nous remercions également notre promoteur monsieur Kameche Abdallah
pour avoir accepté de diriger ce travail, pour son aide, son encouragement et
ses conseils tout au long de ce projet.*

*Nous remercions les membres du jury de nous honorer par leur présence afin
d'évaluer ce travail et pour toutes leurs remarques et critiques.*

*Plus généralement tout le personnel enseignant du département d'informatique
de l'université BLIDA – 1 – qui ont assuré notre formation durant ces 5 années
d'étude ainsi qu'à tous les étudiants qui nous ont aidés de près ou de loin.*

*Un grand merci également à nos familles pour leurs soutiens aussi bien moral
que financier et pour leurs sacrifices.*

MERCI A VOUS TOUS

Résumé

Résumé :

Dans ce travail, nous explorons une approche informatique qui est l'apprentissage par renforcement. Nous explorons principalement les situations d'optimisation d'énergies pour l'internet des objets dans un Maison intelligente et évaluons l'efficacité de différentes méthodes d'apprentissage par renforcement, en évaluant les conceptions par l'analyse mathématique ou par des expériences de calcul, et en exposant les techniques de préservation d'énergie existantes. Nous travaillerons sur le développement d'un modèle de consommation d'énergie au sein Apprentissage par renforcement s'exécutant sur le nœud de détection.

Mots clés : Internet des Objets, Apprentissage par renforcement, Optimisation, consommation énergétique

Abstract

In this work, we explore a computer approach that is learning by reinforcement. We mainly explore energy optimization situations for the Internet of Things in a Smart House and evaluate the effectiveness of different reinforcement learning methods, evaluating designs through mathematical analysis or calculation experiments, and exposing existing energy conservation techniques. We will work on the development of an energy consumption model within Reinforcement Learning running on the detection node.

Keywords: Internet of Things, Reinforcement learning, Optimization, Energy consumption

ملخص

في هذا العمل، نستكشف منهج تكنولوجيا المعلومات هو التعلم المعزز. نحن نستكشف بشكل أساسي مواقف تحسين الطاقة لإنترنت الأشياء في المنزل الذكي وتقييم فعالية أساليب التعلم التعزيزية المختلفة، من خلال تقييم التصميمات عن طريق التحليل الرياضي أو عن طريق تجارب الحوسبة. وفضح تقنيات الحفاظ على الطاقة الحالية. سنعمل على تطوير نموذج لاستهلاك الطاقة في التعلم المعزز يعمل على عقدة الكشف.

الكلمات المفتاحية: إنترنت الأشياء ، تعلم التعزيز ، التحسين ، استهلاك الطاقة

Sommaire

Table des matières

| | |
|-----------------------------|---|
| Introduction générale | 1 |
|-----------------------------|---|

Chapitre 1 : Problème de préservation d'énergie

| | |
|---|----|
| Introduction | 3 |
| 1 Internet des objets..... | 3 |
| 1.1 Historique d'IdO | 3 |
| 1.2 Définition de l'internet des objets | 4 |
| 2 Problématiques posées par l'Internet des objets..... | 6 |
| 2.1 Hétérogénéité de l'Internet des objets..... | 6 |
| 2.2 Influence du monde physique sur l'Internet des objets..... | 7 |
| 2.3 Sécurité..... | 7 |
| 2.4 Protection de la vie privée..... | 9 |
| 2.5 La mobilité | 9 |
| 2.6 Qualité de contexte..... | 10 |
| 2.7 Passage à l'échelle..... | 11 |
| 2.8 Déploiement autonome des entités..... | 12 |
| 2.9 Interopérabilité | 13 |
| 3 Durée de vie d'un RCSF..... | 13 |
| 4 Travaux existants sur l'économie d'énergie dans les RCSF | 13 |
| 4.1 Surveillance de l'énergie à l'aide d'une résistance shunt | 14 |
| 4.2 Estimation d'énergie basée sur logiciel | 15 |
| 4.3 Approche spécialisée basée sur le matériel | 17 |
| 4.4 Estimation de l'énergie basée sur le jeu d'instructions de la CPU..... | 18 |
| 4.5 Modélisation de l'énergie en utilisant une approche paramétrique | 19 |
| 4.6 Boîte noire..... | 20 |
| Conclusion..... | 20 |

Chapitre 2 : Apprentissage par renforcement

| | |
|---|----|
| Introduction | 21 |
| 1 C'est quoi l'apprentissage | 21 |
| 2 Différents types d'apprentissage | 22 |
| 3 Methodologie d'Apprentissage | 23 |
| 3.1 L'apprentissage supervisé | 23 |

| | | |
|-------|---|----|
| 3.2 | L'apprentissage non supervisé | 23 |
| 3.3 | L'apprentissage par renforcement | 24 |
| 4 | Problème d'apprentissage par renforcement | 24 |
| 4.1 | Principe | 24 |
| 4.2 | Formalisation | 25 |
| 4.3 | Processus Décisionnels de Markov | 25 |
| 4.3.1 | Politique | 26 |
| 4.3.2 | Fonctions valeur | 27 |
| 5 | Système d'apprentissage par renforcement | 29 |
| 5.1 | Agent | 29 |
| 5.2 | Le temps | 29 |
| 5.3 | Les états | 30 |
| 5.4 | Les actions | 30 |
| 5.5 | Le signal ou la fonction de renforcement | 30 |
| 5.6 | L'environnement | 31 |
| 6 | Méthodes de résolution | 32 |
| 6.1 | Méthode de la Programmation Dynamique | 33 |
| 6.1.1 | Evaluation d'une politique | 33 |
| 6.1.2 | Amélioration d'une politique | 34 |
| 6.2 | Méthode de Monté Carlo | 35 |
| 6.2.1 | Discussion | 36 |
| 6.3 | Les méthodes différences temporelles (TD) | 37 |
| 6.3.1 | Q-Learning | 38 |
| 7 | Discussion..... | 40 |
| 8 | L'apprentissage par renforcement en profondeur | 41 |
| 8.1 | Apprentissage en profondeur | 41 |
| 8.2 | Le «profond» dans l'apprentissage en profondeur | 41 |
| 8.3 | Comprendre le fonctionnement de l'apprentissage en profondeur, en trois chiffres..... | 43 |
| 9 | Algorithme de rétro propagation du gradient | 45 |
| 9.1 | Principe | 45 |
| 9.2 | Algorithme | 46 |
| 10 | L'apprentissage par renforcement en profondeur | 49 |
| | Conclusion | 50 |

Chapitre 3 : Conception

| | |
|--|----|
| Introduction..... | 51 |
| 1 Architecture générale proposé de la solution | 51 |
| 2 Modélisation de l'Agent Intelligent | 52 |
| 2.1 Concept du Blackjack..... | 53 |
| 2.2 Transposition à notre problème | 53 |
| 3 Serveur | 55 |
| 4 Modélisation UML | 56 |
| 4.1 Diagramme de séquence | 57 |
| 4.2 Diagrammes d'états-transitions | 58 |
| Conclusion | 59 |

Chapitre 4 : Teste et réalisation

| | |
|--|----|
| Introduction..... | 60 |
| 1 Description de l'environnement..... | 60 |
| 1.1 Langage de programmation utilisé..... | 60 |
| 1.1.1 Python | 60 |
| 1.2 Outils de développement | 61 |
| 1.2.1 Spyder..... | 61 |
| 2 Plateforme de test (la solution domotique) | 61 |
| 3 Présentation de l'application..... | 61 |
| 3.1 OpenAI gym environnement..... | 62 |
| 3.2 MC prediction_Sol | 62 |
| 4 Test et validation | 62 |
| Conclusion..... | 64 |
| CONCLUSION GENERALE | 65 |

Liste des figures

| | |
|--|----|
| Figure 1 internet des objets | 5 |
| Figure 2 Estimation d'énergie avec une résistance shunt | 15 |
| Figure 3 Logiciel de calcul d'énergie | 17 |
| Figure 4 Approche spécialisée basée sur le matériel | 18 |
| Figure 5 Instructions de la CPU | 19 |
| Figure 6 Système d'apprentissage par renforcement | 24 |
| Figure 7 Structure des processus stochastique..... | 25 |
| Figure 8 Relation états, actions | 30 |
| Figure 9 Parties d'un système d'apprentissage par renforcement | 32 |
| Figure 10 Un réseau neuronal profond pour la classification des chiffres | 43 |
| Figure 11 Représentations profondes apprises par un modèle de classification numérique | 43 |
| Figure 12 Un réseau de neurones est paramétré par ses poids | 44 |
| Figure 13 Une fonction de perte mesure la qualité de la sortie du réseau | 45 |
| Figure 14 Le score de perte sert de signal de retour pour ajuster les poids | 45 |
| Figure 15 Exemple de réseau MLP à une couche cachée avec 5 entrées, 3 neurones dans la couche cachée, et quatre 4 sorties..... | 47 |
| Figure 16 Apprentissage par renforcement en profondeur..... | 49 |
| Figure 17 Architecture générale | 52 |
| Figure 18 Fonctionnement de Notre Agent | 55 |
| Figure 19 Fonctionnement du Serveur | 56 |
| Figure 20 diagramme de séquence | 58 |
| Figure 21 diagrammes d'états-transitions | 59 |
| Figure 22 graphe de la consommation d'énergie avec 10000 épisodes | 63 |
| Figure 23 graphe de la consommation d'énergie avec 500000 épisodes | 63 |

Liste des tableaux

| | |
|---|----|
| Tableau 1 Algorithme de prédiction des méthodes Différences Temporelles | 38 |
| Tableau 2 Algorithms Q-Learning | 40 |
| Tableau 3 Comparaison des méthodes d'apprentissage par renforcemen..... | 41 |
| Tableau 4 Sommaire de l'algorithme Rétro propagation de l'erreur..... | 49 |

Liste des abréviations

SMES : Systèmes Micro-Electro-Mécaniques

Ido : Internet des objets

QOC : Qualité de contexte

RCSF : Réseaux des Capteurs Sans fil

MDP : Processus de Décision Markovien

Iot : Internet of Things

Introduction générale

Introduction générale

Présentation du sujet :

On assiste aujourd'hui à une nouvelle ère technologique : c'est l'ère de l'après PC (ordinateur personnel). En effet, des ordinateurs portables, des téléphones portables, des PDAs (*Personal Digital Assistant*), des périphériques dotés de GPS (*Global Positioning Systems*), du RFID (*Radio Frequency Identification*) et de l'électronique intelligente devient de plus en plus populaire dans notre vie de tous les jours (Yu et al., 2006).

L'Internet des Objets (IoT – Internet of Things) compte probablement parmi les phénomènes dont on parle le plus de nos jours. L'IoT permet de véritables avancées dans de nombreux domaines tels que la domotique, la santé, les transports, les industries et les villes intelligentes. Toutefois, lors du développement d'un objet connecté, l'autonomie est un des critères les plus importants à tester. Ne pas mesurer la consommation énergétique d'une plateforme IoT peut potentiellement mener à certains problèmes, comme la surchauffe des équipements, la baisse de la durée de vie des batteries, l'augmentation du coût de maintenance de la plateforme, et par conséquent, l'insatisfaction de l'utilisateur.

Au cours de ces dernières années, l'apprentissage des machines est l'un des domaines les plus récents. Pour réaliser cette tâche, il existe plusieurs outils comme la logique floue, les réseaux de neurones artificiels et les algorithmes génétiques qui permettent d'avoir un système de commande intelligent, de plus il existe aussi plusieurs méthodes d'apprentissage, qui sont généralement liées aux sources d'informations utilisées par l'agent. En effet l'agent peut utiliser soit des exemples fournis par un professeur, soit un signal sous forme de récompenses et de punitions qui lui permet de corriger les actions qui a exécuté dans son environnement de travail.

Problématique :

Certaines solutions existent pour améliorer la consommation énergétique, toutefois, beaucoup de scénarios ne peuvent être pris en considération, on peut en citer par exemple les pannes de courant, l'interaction hasardeuse de l'homme avec les équipements ou encore les changements

Introduction Générale

climatiques. De plus, les solutions proposées actuellement sont applicables à une architecture bien spécifique et sont rarement réadaptables à une nouvelle architecture.

Objectif :

Pour remédier à cette problématique, nous nous intéressons à faire une étude comparative des différentes approches existantes, puis proposer une nouvelle solution d'optimisation de consommation énergétique en exploitant l'apprentissage profond par renforcement.

Structuration du mémoire :

Le présent mémoire est composé des parties suivantes :

- Le premier chapitre est consacré à une définition de la notion internet des objets une présentation de quelques problématiques posées par l'IOT. A la fin il y est exposé les travaux existants sur l'économie d'énergie.
- Dans le deuxième chapitre nous allons expliquer le concept d'apprentissage par renforcement et l'apprentissage par renforcement en profondeur et les différentes méthodes utilisés pendant ces processus.
- Le troisième chapitre est réservé à l'étude conceptuelle de notre application ainsi que le rôle de notre agent intelligent.
- La dernière partie nous allons présenter les outils de développement utilisés pour réaliser le système, et l'implémentation de notre système.
- Ce mémoire se termine par une conclusion générale et perspective.

CHAPITRE I :
Problème de
préservation
d'énergie

Introduction

La combinaison de la technologie des réseaux de capteurs avec celle de la SMEM et la nanotechnologie vont certainement réduire sans cesse la taille des nœuds capteurs et amélioreront considérablement leurs capacités en termes de calcul, de capture, de stockage et de bande passante. Mais le problème épineux, celui de l'énergie, reste toujours posé. En effet, chaque nœud capteur est alimenté par une source d'énergie embarquée à capacité limitée (souvent une batterie). Ces nœuds capteurs sont souvent déployés dans des zones hostiles ou dans des milieux où la présence d'êtres humains est impraticable, difficile ou quasi impossible. Ceci exclue dans ce cas toute tentative durable de maintenance exprimée par un rechargement ou un remplacement de ces batteries en mode en ligne dans l'espace de déploiement (Mateu et Moll, 2005).

Dans ce chapitre, on va définir c'est quoi internet des objets et les problématiques proposées par internet des objets, finalement les travaux existants pour la conservation d'énergie pour les Iot domotique, Ces travaux relèvent du développement de techniques intelligentes de gestion de l'énergie au niveau du système.

1 Internet des objets

1.1 Historique d'IdO

L'Internet des objets vise à relier des objets entre eux via un réseau. Chaque objet possède un identifiant unique. Le système d'identification par radiofréquence (RFID) est le plus utilisé dans l'environnement d'IdO.

En fait, il y a quelques développements importants qui ont contribué à la création du paradigme de l'Internet des objets :

L'ordinateur serait de plus en plus "absent", c'est à dire que ces ordinateurs deviennent invisibles dans notre environnement et sont intégrés dans les objets de la vie courante. Il appellera cette notion «ordinateur partout» (où computers everywhere). Puis il la nommera

Chapitre I : Problème de Préservation d'énergie

«informatique ubiquitaire». Le professeur Sakamura a donc montré que l'implantation de puces électroniques dans des objets physiques permettrait de configurer un réseau de capteurs capables de comprendre l'environnement (Sakamura, 2005).

Quatre années plus tard, Mark Weiser, chef scientifique de Xerox Parc a considéré l'ordinateur comme un élément important pour aider l'homme. Cette notion permet de penser à l'Internet des objets.

En 1998, le terme "intelligence ambiante" a été utilisé pour la première fois, par la société Philips. L'intelligence ambiante affirme la volonté de rendre l'informatique invisible en le fusionné dans l'environnement quotidien. Elle regroupe plusieurs idées : l'ubiquité qui se réfère à la présence dans l'environnement de multiples appareils distribués et interconnectés, la sensibilité au contexte qui correspond à collecte et au traitement d'informations concernant l'utilisateur et son environnement, et l'intelligence, qui caractérise la capacité du système à produire des inférences à partir des informations sur l'utilisateur et son environnement.

En 1998, Kevin Ashton, gestionnaire de marque chez Procter et Gamble, a utilisé la notion d'Internet des objets pour la première fois. Il souhaité d'améliorer l'efficacité de la chaîne d'approvisionnement de Procter et Gamble grâce à cette notion. Il s'est intéressé au développement d'un système universel ouvert qui permettrait de connecter des objets à l'Internet (Charline, 2011).

1.2 Définition de l'internet des objets

La notion d'Internet des Objets a été créée en 1999 par Kevin ASHTON en État Unit. On trouve à l'heure actuelle plusieurs définitions pour internet des objets ont été proposées dans la littérature. Par conséquent, il n'y a pas de définition standard pour l'IdO. Les définitions suivantes ont été fournies par différents chercheurs :

L'EPoSS (2008) définit l'origine sémantique de l'expression « internet des objets» qui est composée de deux concepts : « internet » et « objets» où l'internet peut être défini comme un réseau de réseaux informatiques interconnectés, basé sur un protocole de communication standardisé (TCP / IP). Alors que «les objets» sont des objets non identifiables. Par conséquent, «Internet des objets» signifie «un réseau mondial d'objets interconnectés adressables uniquement, basé sur des protocoles de communication standard» (Akyildiz et al.,2005).

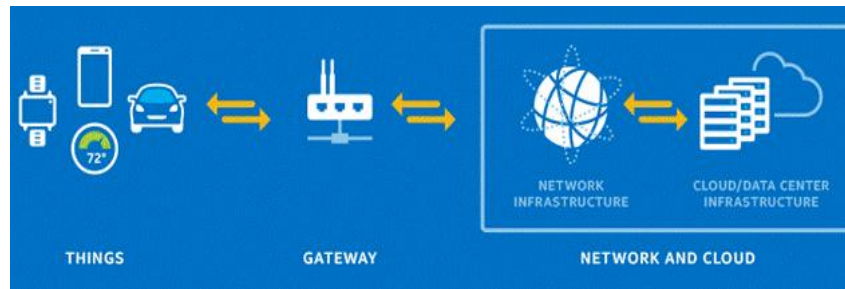


Figure 1 internet des objets(sameh,2016)

Gérald et Sebastian (2008) exprime qu'Internet des objets (IdO) fait partie intégrante d'Internet Future, y compris les développements internet et de réseaux existants. IdO pourrait être défini comme une infrastructure de réseau mondial dynamique avec des capacités de configuration automatique basées sur des protocoles de communication standards et interopérables. Les objets physiques et virtuels possèdent des identités, des attributs physiques et des personnalités virtuelles, utilisent des interfaces intelligentes et sont intégrées au réseau d'information.

IdO sont des objets ayant des identités et des personnalités virtuelles opérant dans des espaces intelligents utilisant des interfaces intelligentes pour se connecter et communiquer dans des contextes sociaux, environnementaux et utilisateurs. Une définition différente, qui met l'accent sur l'intégration transparente, pourrait être formulée comme «Objets interconnectés ayant un rôle actif dans ce que l'on pourrait appeler l'Internet du Futur (**Ovidiu et al., 2011**)

IdO définit comme un paradigme qui évolue régulièrement possédant des centaines de milliards de capteurs et de dispositifs intelligents qui interagissent entre eux sans intervention humaine. Ils généreront une énorme quantité de données, fournissant aux humains l'information en contrôlant les événements et les objets, même dans des environnements physiques distants (**Charith et al., 2014**)

L'Internet des objets (IdO) est le réseau de divers objets physiques intégrés à des logiciels et à divers types de capteurs. La capacité d'interconnexion et de partage de données entre eux pour obtenir plus de valeur et de services. L'intégration d'IdO dans notre vie personnelle peut

certainement apporter de nombreux avantages aux individus, aux entreprises et à la société de diverses manières (Yen-Kuang, 2012).

2 Problématiques posées par l'Internet des objets

2.1 Hétérogénéité de l'Internet des objets

Les propriétés et les capacités des objets varient significativement, contribuant à faire de l'Internet des objets un écosystème certes riche, et aussi très hétérogène. L'IdO est composé d'objets très différents se situent à plusieurs niveaux : matériels, logiciels et protocoles de communication. L'hétérogénéité du système peut venir de différences de protocoles et des réseaux impliqués, de différences d'espace de stockage ou de nature des équipements, ou de variations de la dispersion des entités. Au sein d'un même réseau, les postes de travail dotés de capacités de calcul variés et d'un stockage important peuvent coexister avec d'autres périphériques disposant de ressources limitées. Cette variation nécessite que le middleware de l'IdO ait un niveau d'abstraction plus élevé approprié afin de masquer l'hétérogénéité des modules et la complexité de communication (Benjamin, 2015).

L'hétérogénéité apparaît à deux niveaux : l'hétérogénéité sémantique des objets, et l'hétérogénéité technologique qui demeure une première difficulté à surmonter pour leur permettre d'interagir (Megerian et Potkonjak, 2003) :

- L'hétérogénéité sémantique des objets et des dispositifs : ce type d'hétérogénéité est plus difficile à cause de l'indisponibilité de toutes les entités logicielles lors de la conception. Ce qui contraint de devoir les intégrer dès qu'ils soient disponibles et leurs sémantiques doit être connue. Dans ce cadre, pour résoudre ce problème on doit fournir un mécanisme permettant de connaître la sémantique de différentes entités.
- Hétérogénéité technologique : Les technologies matérielles et logicielles utilisées pour construire les objets sont multiples qui compromettent l'idéal de collaboration autonome entre objets. Afin de rendre les entités technologiquement hétérogènes interopérables, l'utilisation des mécanismes de communication standards est impérativement nécessaire. Ce type d'hétérogénéité nécessite des middlewares distribués sur des serveurs, des stations de travail et des appareils mobiles portables.

2.2 Influence du monde physique sur l'Internet des objets

Le monde physique est un environnement qui évolue naturellement au cours du temps. Dans le paradigme d'IdO, les objets connectés à Internet sont beaucoup plus qu'une simple connectivité ou un message à transporter. En d'autres termes, l'IdO est ouvert, ce qui signifie que tout le monde peut ajouter et supprimer des objets. La capacité de détection des Smartphones et la mobilité des utilisateurs influent sur le monde physique de l'IdO qui nécessite que les développeurs prennent en compte un mécanisme de représentation et de traitement des données. Par conséquent, l'internet des objets est fondamentalement influencé par les caractéristiques du monde physique et des outils qui permettent de le mesurer. Dans ce cas, il s'agira d'adapter les applications aux évolutions du monde qui les entourent.

Nous pouvons diviser cette problématique en trois sous-critères:

- Techniques spécifiques de détection, de correction ou d'atténuation d'erreur: les grands volumes de données issus d'un grand nombre de sources différentes, posent un problème en ce qui concerne leurs mesures qui sont erronées, imprécises ou incomplètes. Il s'agit donc un mécanisme de détection de ces erreurs et leur correction continu en temps réel pour que les applications s'adaptent aux changements qui surviennent au cours du temps de manière pertinente.
- La capacité des objets mobiles à s'adapter aux changements : la mobilité des objets induisent une topologie dynamique du réseau donc ils doivent être en mesure de s'adapter dynamiquement aux changements qui surviennent au cours du temps.
- La capacité des objets mobiles à gérer leurs limites énergétiques : Les applications d'IdO nécessitant un grand nombre des dispositifs qui sont souvent difficiles à mettre en oeuvre en raison des contraintes de temps, de mémoire, de traitement et d'énergie. Le défi d'énergie est largement disponible car les objets sont connectés au réseau. Un aspect de l'IdO va donc consister à réaliser la connectivité internet avec des solutions techniques très basse énergie pour fonctionner avec une simple pile, ou en récupérant uniquement l'énergie dans l'environnement (Benoît, 2012).

2.3 Sécurité

L'internet des objets (IdO) est une intégration de plusieurs réseaux hétérogènes, il devrait traiter des problèmes de compatibilité entre différents réseaux qui est sujette à des problèmes de sécurité. La sécurité représente un élément essentiel pour permettre l'adoption généralisée

Chapitre I : Problème de Préservation d'énergie

des technologies et des applications d'IdO. Les applications de l'IdO peuvent améliorer la vie quotidienne des personnes. Mais, si elles ne peuvent pas assurer la sécurité des informations et de protocole de communication, ces informations peuvent être divulguées à tout moment. Donc, la sécurité de l'IdO ne peut être ignorée. Avec la diffusion volumineuse de l'IdO qui fournira plus vaste riche d'information, le risque d'exposition de ces informations augmentera. Si l'IdO ne peut pas avoir une bonne solution pour les problèmes de sécurité, cela limitera grandement son développement. L'Internet des objets souffre de plusieurs vulnérabilités inhérentes aux technologies utilisées. En effet, de nombreux objets possèdent des capacités matérielles réduites, et de fait, ne peuvent pas directement mettre en oeuvre des techniques de sécurisation modernes (technologie de cryptage, partage de clé.etc). Ces dernières nécessitant des ressources de calcul conséquentes. De la même façon, le coût énergétique élevé des communications sans fils et la puissance de calcul complexifie les processus d'authentification des objets (Benjamin, 2015).

La sécurité peut être mise en oeuvre de deux façons:

- une communication de haut niveau sécurisée entre les pairs de réseau qui permet à la couche supérieure de communiquer entre ses pairs de manière sécurisée et abstraite.
- la gestion sécurisée de la topologie qui traite l'authentification des nouveaux pairs, des autorisations pour accéder au réseau et à la protection des informations de routage échangées dans le réseau (Yeh, 2008).

La sécurité concerne la protection contre les accès non autorisés à une infrastructure, il faut s'assurer que les ressources matérielles et/ou logicielles d'un middleware IdO sont utilisées dans le cadre prévu. Le bloc fonctionnel de sécurité permet de sécuriser le système d'IdO en fournissant des fonctions telles que l'authentification, l'autorisation, la confidentialité, l'intégrité, la sécurité des données, et de la non-répudiation : La confidentialité indiquant la garantie que seules les entités autorisées peuvent accéder aux données et les modifier comme l'utilisation des techniques de cryptage. L'intégrité garantit que les informations ne sont pas modifiées ou corrompues (Jing, 2014).

2.4 Protection de la vie privée

Dans la vision d'IdO, les technologies de communication sans fil joueront un rôle important. L'adoption omniprésente du support sans fil pour l'échange de données peut poser un nouveau problème en matière de violation de la vie privée. En fait, le canal sans fil augmente le risque de violation en raison des capacités d'accès à distance, ce qui expose potentiellement le système à des attaques d'écoute. La vie privée représente donc un véritable problème qui peut limiter le développement de l'IdO. Pour assurer la protection de la vie privée dans le paradigme d'IdO, il faut définir un ensemble de règles et de protocoles qui définissent quels dispositifs ou quelles applications ont le droit d'accéder aux informations de contexte (Malek, 2014).

La protection de la vie privée doit être traitée à différents niveaux:

Au niveau le plus bas, la couche matérielle doit assurer la sécurité et la confidentialité pendant la collecte et le stockage temporaire dans l'appareil.

- Au niveau communication ou les protocoles sécurisés doivent s'assurer que la communication est bien protégée.
- Au niveau application : Une fois les données reçues, la protection au niveau de l'application doit être en place pour surveiller et contrôler qui peut voir ou utiliser ces données (Charith et al., 2014).

2.5 La mobilité

Une vision cohérente et stable de la topologie du réseau est un facteur important pour la gestion des applications d'IdO. Cependant, la mobilité liée à tout (objets, utilisateur.etc) rend cette cohérence difficile et inefficace (Jun, 2015). Parce que les objets intelligents dans l'IdO ne sont pas statiques et changent leur emplacement physique, cela entraîne des changements rapides dans la topologie du réseau. Ce changement à son tour affecte l'efficacité du routage car de nombreux chemins deviennent indisponibles. La mise à jour des informations de routage après la mobilité des objets peut entraîner des coûts généraux importants, notamment lorsque de nombreux noeuds ont changé leur emplacement physique simultanément. De nombreuses applications de contexte sont également mobiles et dépendent des sources

d'informations mobiles. Donc il faut un mécanisme pour que le provisionnement de l'information de contexte doive être adaptable à l'environnement changeant (Willig, 2006).

2.6 Qualité de contexte

Dans l'Internet des objets (IdO), l'un des nombreux facteurs importants est la sensibilité de contexte. Mais les informations de contexte peuvent ne pas être fiables ou utiles, devenant un problème en termes de qualité. Par conséquent, un aspect important est que ces informations de contexte doivent être fiables et la qualité doit être assurée. Dans ce nouvel paradigme, les données collectées à partir d'un déploiement mondial d'objets intelligents constituent la base pour prendre des décisions intelligentes et fournir des services aux utilisateurs.

Les limites des différentes sources d'information, la qualité de la communication et du traitement des données, et la situation d'une mesure spécifique peuvent affecter sur la qualité des informations de contexte collectées. Les applications d'IdO doivent traiter d'énormes volumes de données, avec une grande vélocité, dynamicit  et une grande vari t  de types de donn es. De plus, divers applications ont des exigences diff rentes qui d termineront l'efficacit  de l'utilisation des donn es. Si les donn es sont de mauvaise qualit  ou ils ne sont pas facilement utilis  et compris par l'utilisateur, les d cisions risquent d' tre non pertinentes et m ne vers une d gradation des performances du syst me sensible au contexte (Jean-Paul et al., 2014).

Les sources de contexte dans l'IdO peuvent  galement avoir des capacit s diff rentes pour observer l'environnement qui offrent le m me type d'information avec des diff rentes qualit s fonctionnelles et non fonctionnelles. L'un des principaux d fis auxquels sont confront s les d veloppeurs pour r aliser la vision d'infrastructure d'IdO est de s lectionner la source de contexte le plus appropri  disponible qui a une meilleur qualit  et ad quate au besoin de l'application. La gestion de la QoC au sein des gestionnaires de contexte doit donc s'effectuer tout le long du cycle de vie des informations, depuis leur collecte jusqu'  leur acheminement vers les applications.

Par cons quent, on peut diviser ce probl me en des sous-probl mes comme suit:

- QoC pour les informations de contexte de la phase d'acquisition : évaluer la qualité des informations produites au niveau des sources d'informations peut être réalisée par analyser les caractéristiques des sources de contexte.

QoC pour les informations de contexte de la phase de traitement : dans ce niveau, évaluer la qualité de contexte consiste à augmenter le niveau d'abstraction des informations de QoC. Prend en compte l'évaluation dans ce niveau permet d'éliminer les informations de contexte qui ont la QoC est mal et faible et donc augmenter la performance dans le processus de traitement des informations.

- QoC pour les informations de contexte de la phase de présentation : dans ce niveau, l'information de contexte avec sa qualité sont présentés aux applications selon leurs besoins pour suggérer de bonnes décisions aux utilisateurs finaux ([Pierrick, 2015](#)).

2.7 Passage à l'échelle

L'évolution rapide de la technologie d'IdO définit de nouveaux défis d'architecture en termes de passage à l'échelle. Ce nouvel paradigme est composé d'un nombre croissant d'objets intelligents, ces objets communiquent ensemble. Le middleware proposé devrait conserver sa stabilité et devrait avoir la capacité de gérer efficacement le nombre croissant d'objets et d'exécuter leurs fonctions avec l'efficacité requise, bien que le nombre d'appareils connectés augmente et varie d'un endroit à l'autre. Dans ce cas, une architecture décentralisée est requise qui distribue les informations provenant de différents d'objets d'IdO afin de ne pas se retrouver dans une situation de goulot d'étranglement ([Ghofrane, 2015](#)).

Les différents problèmes liés au problème de passage à l'échelle sont résumés comme suit :

- Adressage et nommage : En raison de la croissance rapide des appareils d'IdO, cela implique un espace d'adressage important pour identifier ces objets et la découverte de ces derniers.
- Communication de données et réseaux : En raison du haut niveau d'interconnexion entre de nombreuses entités qui nécessitent un réseau mondial d'objets uniquement adressable basé sur des protocoles de communication standard.
- La gestion de l'information et des connaissances : différents types d'objets hétérogènes génèrent des données en fonction de leurs lectures ou de la communication avec d'autres objets. Une énorme quantité de données continuera à générer à partir de

différentes parties du monde d'IdO et cela nécessite une bonne gestion du stockage et de traitement pour faire face à un tel volume de données.

- Gestion des services: en raison du nombre important de services qui pouvant être disponibles dans l'IdO, cela implique une bonne gestion des ressources hétérogènes ([Daniele et al., 2012](#)).

2.8 Déploiement autonome des entités

Le déploiement est un processus complexe qui a pour objectif la mise à disposition puis le maintien en condition opérationnelle d'un système logiciel. Le déploiement implique deux catégories de sites: un site producteur qui héberge des composants logiciels et des procédures d'installation, et un site consommateur qui est la cible du déploiement, sur lequel un élément logiciel doit être exécuté. Tous ces sites sont des domaines de déploiement nécessitant l'écriture de la distribution des composants, ce que l'on appelle un plan de déploiement (association entre les composants logiciels et les appareils).

Dans le paradigme d'IdO, un gestionnaire de contexte est un élément essentiel pour traiter les informations de contexte. Ce gestionnaire de contexte est un système de composants répartis qui doivent être déployés sur différents appareils : des appareils pour l'interaction avec l'utilisateur, des appareils (serveurs) pour traiter les informations collecté, les appareils pour produire et collecte les informations.

Les solutions traditionnelles de déploiement consistent à utiliser le mode centralisé dans lequel un opérateur humain réalise les différentes tâches du déploiement. Ces solutions utilisent dans la majorité des cas pour les environnements qui sont limités et leurs composants sont connus à l'avance. Par conséquent, ces solutions de déploiement sont globalement inadaptées et impossible à réaliser dans l'environnement d'IdO du fait de l'hétérogénéité, de la dynamique, de l'ouverture et de la décentralisation de ce système. Ainsi, dans l'IdO, le réseau de machines sur lesquelles on fait le déploiement n'est pas connu à l'avance du fait de l'apparition et la disparition d'appareils, ce qui rend la tâche du déploiement difficile à réaliser ([Raja, 2015](#)).

2.9 Interopérabilité

L'Internet des Objets est constitué de dispositifs hétérogènes qui interagissent et collaborent entre eux pour réaliser une tâche commune et pour échanger les informations. Comme un nombre croissant d'objets sont ajoutés à l'internet des objets, l'interaction de ces composants ou plusieurs systèmes pour échanger les informations devient nécessaire à réaliser. Le middleware qui sera utilisé dans tels cas devrait être aussi interopérable que possible afin qu'il puisse supporter les objets hétérogènes existants ainsi que d'autres nouveaux objets intelligents qui peuvent se produire dans le futur.

Deux types d'interopérabilité possibles peuvent être atteints :

- D'un côté, les composants internes peuvent interagir entre eux et partager les informations.
- D'un autre côté, différents middlewares peuvent communiquer entre eux et utilisent les informations échangées ce qu'on appelle l'interopérabilité externe (Xin, 2015).

3 Durée de vie d'un RCSF

Pour un réseau de communication classique, qui est conçu en général pour supporter plusieurs utilisateurs individuels, la durée de vie est sujette à de nombreuses interprétations. Un réseau de communication peut être considéré non opérationnel (ou déclaré mort) par un utilisateur alors qu'il continue à fournir une qualité de service pour d'autres utilisateurs. Au contraire, un RCSF n'est pas déployé en tant qu'ensemble de nœuds individuels, mais en tant qu'ensemble de nœuds capables d'exécuter des tâches collaboratives spécifiques au, niveau du réseau. C'est pour cette raison que la durée de vie dans ce cas a une définition non ambiguë : c'est le temps écoulé entre l'instant de déploiement et l'instant où le réseau devient incapable de réaliser correctement sa tâche à laquelle il a été dédié. Nous allons revenir à cette notion dans le prochain chapitre pour voir qu'il existe plusieurs définitions proposées dans la littérature (Farida, 2019).

4 Travaux existants sur l'économie d'énergie dans les RCSF

La consommation d'énergie d'un appareil intégré peut être acquise à l'aide de divers

Approches. Cependant, nous avons globalement classé ces méthodes en mode hors connexion et approches en ligne et approches logicielles et matérielles. Dans l'approche en ligne, la consommation d'énergie est estimée au moment de l'exécution, alors que dans l'approche hors ligne la consommation d'énergie est estimée en laboratoire avant de déployer le nœud dans le réseau. De même, dans l'approche basée sur le matériel, un logiciel spécialisé matériel est utilisé pour mesurer la consommation d'énergie, cependant, dans le logiciel approche, le logiciel d'application est en cours de modification pour estimer son consommation. Les approches logicielles et matérielles peuvent être utilisées dans les approches en ligne et hors ligne pour prévoir le coût de l'énergie.

Au cours des dernières années, de nombreuses recherches ont été menées sur le profilage énergétique des réseaux mobiles. plates-formes informatiques. L'utilisation efficace de l'énergie par ces dispositifs informatiques est un facteur clé pour les fabricants d'appareils et les développeurs d'applications. Depuis ces Les appareils portatifs fonctionnent également sur piles et la capacité de ces appareils est faible. strictement limité aux contraintes de poids et de taille de l'appareil. Ainsi, il est essentiel pour que ces appareils gèrent de manière optimale leur consommation d'énergie ([Armen et al., 2016](#)).

Dans les suivantes, nous aborderons certaines des techniques existantes utilisées pour mesurer la puissance et la consommation d'énergie des appareils mobiles, des nœuds de capteurs sans fil et autres appareils intégrés. Puisque tous ces appareils fonctionnent avec une batterie restreinte

4.1 Surveillance de l'énergie à l'aide d'une résistance shunt

[Carroll et Heiser \(2010\)](#) ont effectué des tests rigoureux, puis simulé plusieurs scénarios d'utilisation pour montrer l'importance de la puissance consommée par divers composants: CPU, mémoire, écran tactile, matériel graphique, audio, stockage et interfaces réseau un appareil mobile. Pour le profilage énergétique, ils ont mesuré la consommation électrique physique de chaque composant. Ils ont inséré une résistance de détection sur le rail d'alimentation pour le composant concerné et ont utilisé la loi d'ohm ($V = I * R$) pour mesurer la consommation de courant. [Rice et Hay \(2010\)](#) ont mené une étude sur la consommation d'énergie des périphériques connectés aux réseaux sans fil 802.11. Pour mesurer la consommation d'énergie, ils ont utilisé une approche similaire à celle de Carroll et

Chapitre I : Problème de Préservation d'énergie

Heiser. Ils ont inséré une résistance de dérivation en série avec l'alimentation et le téléphone portable pour mesurer la consommation d'énergie globale de l'appareil mobile. Ils ont utilisé un 0,02 haute précision résistance en série avec borne de batterie et son connecteur sur le téléphone. Dans les deux approches, ils ont utilisé une résistance de dérivation en série avec le composant et l'alimentation pour mesurer la consommation de courant. Ces deux approches entrent dans la catégorie de l'approche hors ligne pour estimer la consommation d'énergie.

la mesure de la consommation d'énergie au moment de l'exécution était essentielle pour les études liées à l'optimisation de la puissance cible (Milenkovic et al., 2005). Ils ont proposé deux approches pour mesurer la consommation d'énergie d'un nœud dans un réseau de capteurs sans fil. Dans la première approche, ils ont échantillonné l'alimentation et les tensions de sortie à l'aide d'une sonde de courant. Dans la deuxième approche, ils ont échantillonné l'alimentation et la tension à la résistance de shunt. une méthode permettant de mesurer en temps réel la consommation d'énergie par un réseau WBAN (Wireless Body Area Network). Ils ont utilisé une résistance shunt en série avec l'alimentation pour mesurer la consommation de courant globale d'un nœud sans fil (Mijovic et al.,2015).



Figure 2 Estimation d'énergie avec une résistance shunt (Michel et Ianoz, 2004)

4.2 Estimation d'énergie basée sur logiciel

Proposition d'une approche de modélisation de la consommation d'énergie basée sur un appel système. Ils ont proposé d'estimer la consommation d'énergie en fonction des appels système envoyés au système d'exploitation. Quand une application veut accéder un pilote d'E / S ou de mémoire cet appel passe par un système d'exploitation et le modèle proposé peut estimer la quantité d'énergie consommée une fois cet appel terminé. Ce modèle effectue une estimation de la puissance au moment de l'exécution et fournit une consommation d'énergie fine et

Chapitre I : Problème de Préservation d'énergie

détaillée au niveau de l'application. Nous avons classé ces deux approches dans la zone de la méthode hors ligne basée sur un logiciel pour prédire la consommation d'énergie.

Li et John (2003) ont développé le profil énergétique d'un système d'exploitation en utilisant un large éventail d'applications. Ils ont proposé divers modèles pour estimer efficacement la dissipation d'énergie au moment de l'exécution d'un système d'exploitation.

Proposition d'une méthode automatisée de mesure de l'énergie pour les applications fonctionnant sous Android mobile et bar métal dispositifs informatiques embarqués. Ils ont exploité différents aspects matériels et logiciels de l'application et plusieurs autres approches pour effectuer une mesure de la puissance au moment de l'exécution. Pour le profilage énergétique, ils ont utilisé un simulateur de batterie fourni par National Instruments. Le simulateur de batterie fournit un échantillonnage discret, à haute résolution et à haute fréquence du courant consommé par le dispositif informatique.

Proposition des modèles basés sur les flux et sur le temps pour mesurer la consommation d'énergie par un nano data center (nDC). Les nDC sont des serveurs hautement distribués qui fonctionnent homologue pour héberger et transmettre du contenu et des applications. Les nDC deviennent de plus en plus populaires pour être utilisés comme serveurs locaux pour les services IoT. Dans le flux modèle, la consommation électrique de l'équipement est calculée par rapport à l'ensemble des flux de données transitant par l'équipement. De même, dans le modèle temporel, la consommation d'énergie est mesurée en termes de temps qu'un équipement consacre à la réalisation d'un service cloud (**Jalali et al.,2016**).

Une méthodologie générale sur la manière de modéliser la consommation d'énergie d'un nœud dans un réseau sans fil à une étape pré-déployée ou pré-production. Dans cette approche, ils ont adopté une approche système et pris en compte tous les processus consommateurs d'énergie, tels que les capacités de mise en réseau, de détection, de traitement et d'acquisition, pour estimer la consommation électrique d'un nœud.

présenté un environnement de simulation évolutif permettant d'estimer une consommation d'énergie précise par nœud dans un réseau de capteurs sans fil (**Shnayder et al.,2004**).

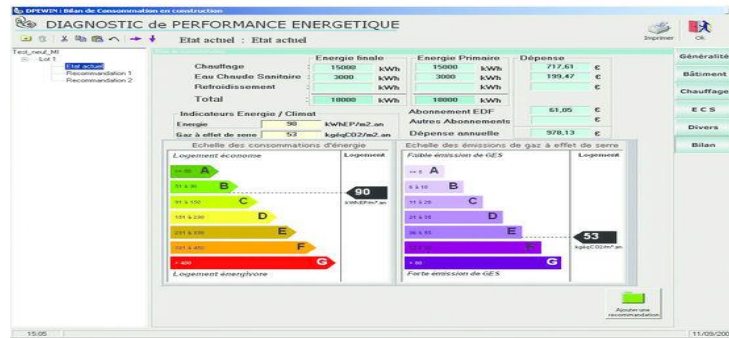


Figure 3 Logiciel de calcul d'énergie (Humane et al., 2015)

4.3 Approche spécialisée basée sur le matériel

(Homb, 2016) a proposé une approche d'exécution pour mesurer la consommation d'énergie à l'échelle du système à l'aide d'un matériel spécialisé. Il a effectué des mesures en temps réel du courant et de la tension de l'appareil cible et a stocké les données sur une carte SD. Ces données ont ensuite été transférées manuellement sur un ordinateur pour une analyse plus approfondie. Le seul inconvénient de cette approche est de traiter une grande quantité de données, ce qui ajoute à la consommation énergétique globale du périphérique.

Utilisé l'approche consistant à utiliser un matériel spécialisé pour fournir la consommation d'énergie. Un matériel intégré connecté au périphérique intégré cible et un logiciel associé pour fournir les données de consommation électrique. Ils ont utilisé ARM7TDMI comme matériel intégré pour effectuer des mesures d'énergie à bord. Leur modèle estime les applications d'une application. consommation d'énergie de manière hybride. La consommation d'énergie du cœur de l'unité centrale est directement mesurée par l'outil intégré. Toutefois, pour estimer la consommation d'énergie de la mémoire, ils ont proposé un modèle de mémoire. Dans ce modèle, ils prédisent la consommation d'énergie de la mémoire en utilisant les traces de mémoire collectées des consommations d'énergie de la mémoire du système. Dans cette méthode, ils ont utilisé les quatre approches; hors ligne, en ligne, logiciel et matériel, pour estimer la consommation d'énergie.

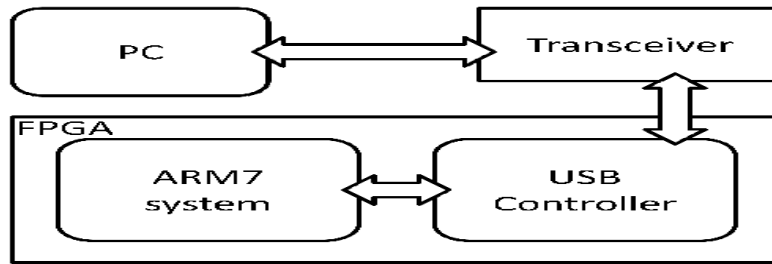


Figure 4 Approche spécialisée basée sur le matériel (Wolciech et Oleksiak, 2015)

4.4 Estimation de l'énergie basée sur le jeu d'instructions de la CPU

Proposé l'utilisation de la simulation de code statique et du modèle de puissance du microprocesseur pour prédire la consommation énergétique d'un programme exécuté sur un périphérique intégré ([Acevedo et al.,2010](#)). Leur modèle prend en compte trois types de coûts énergétiques; le coût d'exécution de l'instruction elle-même, le coût de commutation entre les instructions, et le coût de la mauvaise prédiction de cache-miss et de branche. Il s'agit d'un processus ponctuel permettant d'estimer la puissance et l'énergie des applications exécutées sur un microprocesseur. Ce processus doit être exécuté à nouveau si la micro-unité de commande (MCU) a été modifiée. Dans leur travail, ils ont montré qu'une erreur d'estimation de l'énergie était de 7% et 14,6% respectivement. Cette approche convient à l'analyse de code statique. De même, cette approche est également spécifique pour estimer le coût du processeur lui-même et ignore la consommation d'énergie des périphériques.

Suggéré un modèle d'estimation de l'énergie au niveau instruction pour les systèmes embarqués ([Bazzaz et al.,2013](#)). Ils ont utilisé un microcontrôleur basé sur ARM7TDMI pour déterminer les paramètres du modèle. Dans leur modèle, l'énergie totale consistait en la consommation d'énergie du cœur du processeur, de la mémoire Flash, du contrôleur de mémoire et de la mémoire SRAM. Leur Les résultats expérimentaux reposaient sur plusieurs applications intégrées de la suite de tests de performances et montraient que l'erreur d'estimation était inférieure à 6%.

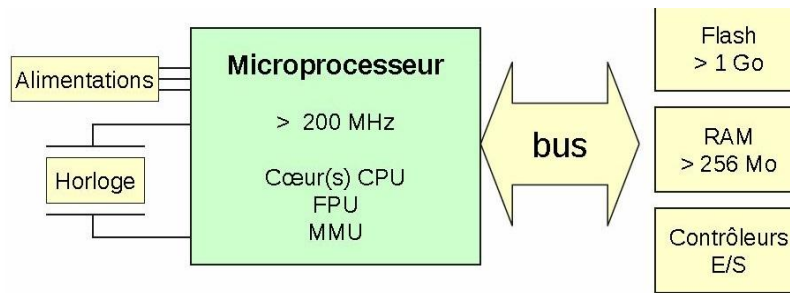


Figure 5 Instructions de la CPU (Kurowski et al., 2013)

4.5 Modélisation de l'énergie en utilisant une approche paramétrique

Proposé une technique utilisant les fonctions fonctionnelle et paramétrique modélisation du processeur pour estimer la consommation d'énergie au moment de l'exécution ([Laurent et al.,2004](#)). Dans leur approche, ils ont extrait les paramètres importants de l'application cible et plate-forme qui a contribué de manière significative à la consommation d'énergie. Puis ils développé des programmes d'assemblage élémentaires (appelés scénarios) et mesuré leur puissance et leur consommation d'énergie. Enfin, ces valeurs ont été utilisées pour développer une modèle de régression pour estimer la consommation d'énergie pour différentes applications et plates-formes matérielles. Ils ont utilisé l'approche logicielle hors ligne pour créer un modèle d'estimation d'énergie. Dans leur approche, l'erreur moyenne d'estimation varie sur différentes applications et processeurs.

Les modèles paramétriques basés sur l'architecture et l'algorithme d'estimation de la consommation d'énergie du traitement du signal numérique Applications. Les erreurs d'estimation de leur modèle sont de 7,1% et 7,6% pour les impulsions finies. Réponse (FIR) fonctionnant sur les processeurs C6701 et C5501 respectivement. De même, ils ont observé des erreurs d'estimation de 4% et 6,6% pour les applications qui transformées de Fourier rapides (FFT) calculées.

Les compteurs de performance du processeur puissent être utilisés pour mesurer la consommation d'énergie du processeur. Ils ont utilisé la performance liée les événements au sein d'un microprocesseur comme les erreurs de cache et les transactions DMA pour déterminer consommation d'énergie dans la mémoire, le disque et d'autres sous-systèmes en dehors du microprocesseur.

Chapitre I : Problème de Préservation d'énergie

Sur la base de la matrice de performance, ils ont développé des modèles spécifiques au système pour mesurer la consommation de six sous-systèmes: microprocesseur, unité de traitement graphique (GPU), chipset, mémoire, E / S et disque. Leurs résultats ont montré que ces modèles produisent une erreur moyenne inférieure à 9% par sous-système en considérant différents charges de travail.

4.6 Boîte noire

Suggéré une technique d'estimation de l'énergie permettant de modéliser consommation d'énergie sous diverses combinaisons d'états de périphériques sans utiliser de matériel supplémentaire. Ils ont utilisé la consommation électrique nominale des spécifications des dispositifs associés pour estimer la consommation d'énergie. La technique de s'appuyer sur la spécification de la fiche technique pour l'estimation de l'énergie est appelé boîte noire l'estimation de l'énergie et convient aux cas où vous ne pouvez connecter aucun réseau externe. Matériel pour mesurer la consommation d'énergie ([Hawng et al., 2010](#)).

Conclusion

Grâce à leur immense potentiel applicatif, les RCSF ont suscité beaucoup d'intérêt ces dernières années de la part des professionnels du domaine des réseaux, les industriels et même les décideurs. Bien que la technologie des RCSF a connu des avancées sensibles sur les aspects théorique notamment, certains problèmes importants comme par exemple la consommation d'énergie restent toujours posés.

Dans ce chapitre, après nous avons parlé sur l'internet des objets, objet de notre étude. Ce chapitre concerne principalement les concepts de base, Hétérogénéité et l'influence du monde physique sur l'Internet des objets.

A la fin de ce chapitre, nous avons évoqué une classification des travaux proposés ces dernières années sur l'économie d'énergie parmi ces techniques nous intéressons l'approche paramétrique car utilise l'apprentissage par renforcement pour résoudre le problème d'énergie donc c'est notre objectif pour l'optimisation de la consommation d'énergie dans un contexte IoT.

CHAPITRE II :
Apprentissage par
renforcement

Introduction

Le but initial de l'intelligence artificielle (IA), domaine né dans les années 50, était de créer des systèmes informatiques qui égaleraient voire devanceraient l'homme dans de nombreuses activités réputées intelligentes comme raisonner, résoudre des problèmes complexes ou naviguer pour accomplir une mission dans un environnement totalement inconnu tout en s'adaptant au changement de ce dernier. Les résultats n'ayant jamais atteint ces espérances, ce domaine a aujourd'hui pour objectif la compréhension et l'adaptation des mécanismes immanents du comportement "intelligent" pour les appliquer aux systèmes informatiques.

La notion d'agent situé ou isolé est une approche récente pour aborder la modélisation et la résolution de problèmes en utilisant l'idée que l'on peut automatiquement apprendre en interagissant avec son environnement : C'est le concept fondamental de l'intelligence artificielle située (IAS). Son avantage est qu'il offre un bon cadre pour la structuration des informations: les données contenues dans l'agent représentent les objectifs visés et celles intégrées dans l'environnement représentent les ressources externes dont dispose ce dernier. Avec cette boucle sensori-motrice, on arrive finalement à intégrer le raisonnement au sein de cette entité.

Dans notre étude, nous nous sommes intéressés à cette dernière méthode qui est appelée *apprentissage par renforcement*. Dans l'apprentissage par renforcement l'agent ne dispose plus d'exemples de commande ou de trajectoires de référence. L'apprentissage est réalisé grâce à une évaluation des résultats.

Ce chapitre est consacré pour la présentation des types d'apprentissage qui existent, ainsi pour la description de la technique d'apprentissage par renforcement et ces éléments essentiels et de l'apprentissage par profond (*deep Q-Learning*).

1 C'est quoi l'apprentissage

D'une manière très générale, on définit l'apprentissage comme étant l'ensemble de règles de modification du système en fonction des données observées (Munos, 2007).

Chapitre II: Apprentissage Par Renforcement

C'est aussi « la capacité à faire mieux la prochaine fois » (Dutech, 1999). Les termes utilisés dans cette définition sont assez flous et exigent une signification plus explicite.

- **Faire mieux** : Soit que le système devient plus efficace, plus rapide, soit qu'il capitalise sur l'augmentation de ses connaissances et de ses capacités à raisonner. Ainsi, un système apprend s'il améliore sa capacité à résoudre des problèmes en utilisant de l'information obtenue par interaction sur son environnement ou bien s'il construit ou modifie une représentation interne de ses expériences.
- **La prochaine fois** : la majeure partie des travaux de recherche utilise des expériences qui sont exactement reproductibles dans le temps. La prochaine fois laisse sous entendre que le système devra savoir reconnaître des problèmes et des situations semblables pour pouvoir les intégrer à sa représentation du monde sans avoir à traiter chaque cas comme un cas particulier.

2 Différents types d'apprentissage

L'apprentissage peut être centré selon quatre axes : agent, environnement, interaction ou organisation (Vercoouter et al. 1998).

- a) **Apprentissage centré agent** : Concerne ce qu'un agent peut apprendre sur lui-même ou sur les autres agents. Cet apprentissage porte sur le comportement de l'agent, ses stratégies et ses décisions
- b) **Apprentissage centré environnement** : se focalise sur ce que l'environnement peut apprendre sur l'agent. Cela peut porter sur nombreux objets tant la diversité de l'environnement peut être grande. Dans un environnement à forte dynamique, un agent peut apprendre sur les parties nouvellement apparues (vie artificielle)
- c) **Apprentissage centré interaction** : porte sur les moyens mis en œuvre par les agents pour communiquer ou interagir (nouveaux langages, communications.)

- d) **Apprentissage centré organisation** : s'occupe de faire évoluer les rôles des agents au sein de leur société. Lorsque le système est supervisé, les agents sont guidés dans leur démarche, les agents autonomes quant à eux doivent apprendre à s'auto-organiser (vie artificielle).

3 Méthodologie d'Apprentissage

L'apprentissage est un processus visant à améliorer les performances d'un système en se basant sur ses expériences passées. Cette méthode intervient lorsque le problème paraît trop compliqué à résoudre en temps réel, ou lorsqu'il paraît impossible de résoudre le problème de manière classique et rigoureuse.

Il existe trois grandes familles de méthodes d'apprentissage: l'apprentissage supervisé, l'apprentissage non supervisé, et l'apprentissage par renforcement.

3.1 L'apprentissage supervisé

C'est l'apprentissage le plus couramment utilisé et il est très bien maîtrisé. Son inconvénient est que l'agent n'est pas immédiatement autonome, puisqu'il a besoin d'un superviseur qui dans un premier temps lui indique la marche à suivre dans des situations qu'il pourra rencontrer. Si la base d'apprentissage est complète, l'agent saura réagir aux situations auxquelles il sera confronté.

Cependant les situations doivent présenter une certaine constance : si l'agent est confronté à une situation entièrement nouvelle, il sera incapable de s'y adapter car aucun exemple donné par le superviseur n'y correspondra.

3.2 L'apprentissage non supervisé

Contrairement à l'apprentissage supervisé, on ne fournit ici qu'une base d'entrées, et c'est le système qui doit déterminer ses sorties en fonction des similarités détectées entre les différentes entrées (règle d'auto organisation).

3.3 L'apprentissage par renforcement

Imaginant un agent, capable de percevoir et agir, qui va choisir ses actions en fonction de sa perception de son environnement et dont le but va être de maximiser une récompense qu'il obtient en fonction des actions qu'il réalise (Camille, 2005).

Dans ce contexte, l'apprentissage par renforcement est une technique qui permet de trouver par un processus d'*essais et d'erreurs*, l'action optimale à effectuer pour chacune des situations « états » que l'agent va percevoir afin de maximiser ses récompenses (David, 2004).

C'est une méthode d'apprentissage *orienté objectif* qui conduit à un contrôleur optimal pour la tâche spécifiée par les récompenses. Elle est aussi non supervisée car la récompense ne donne pas l'action optimale à réaliser mais simplement une évaluation de la qualité de l'action choisie (David, 2004).

4 Problème d'apprentissage par renforcement

4.1 Principe

L'apprentissage par renforcement définit un type d'interaction entre l'agent et l'environnement à partir d'un état ou une situation s dans l'environnement, l'agent choisit et exécute une action a qui provoque une transition vers l'état s' . Il reçoit en retour un signal de renforcement r . Ce signal est utilisé par l'agent pour améliorer sa stratégie « politique », c'est-à-dire la séquence de ses actions, afin de maximiser ses récompenses futures (Youssef, 2004).

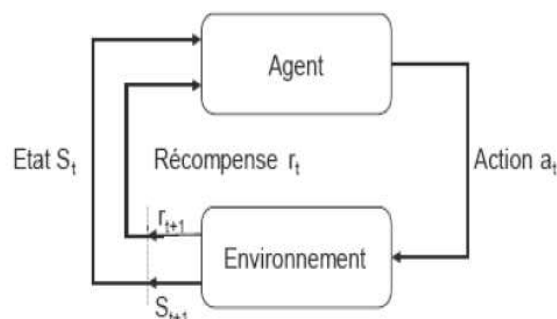


Figure 6 Système d'apprentissage par renforcement (Mezaache, 2008)

Chapitre II: Apprentissage Par Renforcement

4.2 Formalisation

Pour un agent le problème de l'apprentissage par renforcement est défini par l'utilisation des éléments suivants :

- Un ensemble d'états S correspondant à la perception de l'agent à son environnement.
- Un ensemble d'actions possibles A .
- Une fonction de renforcement R .

Le but de l'agent est de choisir les actions qui lui amènent le plus de récompenses et le moins de punitions.

Ceci nous conduit à donner une formalisation pour ce problème d'apprentissage en utilisant un Processus de Décision Markovien (MDP).

4.3 Processus Décisionnels de Markov

Le Processus Décisionnel de Markov forme le modèle le plus classique des problèmes de décision séquentielle. C'est un processus stochastique, il assigne des récompenses aux transitions des états.



Figure 7 Structure des processus stochastique

Chapitre II: Apprentissage Par Renforcement

Les récompenses et les probabilités de transitions d'un état à un autre sont des fonctions de l'état courant et de l'action courante, et non de la trajectoire passée « états précédents, actions précédentes ».

Un Processus de Décision Markovien « MDP » est un modèle de Markov d'ordre 1 on le définit par :

- S : c'est l'espace d'états.
- A : c'est l'espace d'actions.
- T : c'est une fonction de transition markovienne qui est défini de $S \times A \times S \rightarrow [0, 1]$. (s, a, s') représente la probabilité d'aller de l'état « s » à l'état « s' » en effectuant l'action a .
- r : c'est une fonction des renforcements Markovienne qui est défini de $r(s, a, s')$ représente le renforcement obtenu en effectuant l'action « a » et en se retrouvant dans l'état « s' ».

Remarque

On dit que la fonction de Transition est Markovienne car la probabilité de transition entre s et s' , lorsqu'en effectue l'action a ne dépend pas des états précédents et/ou des actions précédemment effectuées. On dit aussi que la fonction de renforcement est Markovienne car le renforcement reçu lors de la transition entre s et s' lorsque on effectue l'action a ne dépend pas des états précédemment visités et/ou des actions précédemment effectuées ([Pascal, 2004](#)).

4.3.1 Politique

Une politique c'est la stratégie ou la façon qui modélise ou représente le comportement d'un agent dans un instant t dont il va choisir une action « a » à exécuter dans un état « s » donné. Dans le cas général il s'agit d'une fonction déterministe ou aléatoire qui est notée par π . Une politique aléatoire permet de choisir pour un état donné différentes actions suivant une probabilité, elle est défini de l'ensemble des couples état-action dans l'intervalle $[0, 1]$.

On écrit :

$$\pi: S \times A \rightarrow [0, 1]$$

$$(s, a) \rightarrow \pi(s, a) = \Pr(a_t = a, s_t = s).$$

La politique π nous donne la probabilité de choisir une action donnée.

Une politique déterministe ne pose qu'une action par état elle est définie par :

$$\pi: S \rightarrow A .$$

$$s \mapsto \pi(s) = a .$$

La politique est le cœur du processus de l'apprentissage par renforcement puisqu'elle est suffisante pour déterminer le comportement de l'agent.

4.3.2 Fonctions valeur

C'est une fonction d'estimation où la plupart des problèmes d'apprentissage par renforcement sont basés sur elle. Elle est définie pour un *état* ou un pair « *état, action* », cette fonction permet à l'agent de savoir comment il apprend à choisir les bonnes actions et comment mesurer leurs utilités.

Une fonction valeur est optimale si elle est égale à la somme pondérée des renforcements reçus depuis l'état initial jusqu'à l'état final en exécutant des actions optimales.

4.3.2.1 Fonction « Valeur - état »

La valeur d'un état $s \in S$ en appliquant la politique π est notée par v , c'est une fonction des états dans les réels \mathbb{R} , $V^\pi : S \rightarrow \mathbb{R}$. Elle est définie comme suit :

$$V^\pi(s) = E_\pi\{R_t / s_t = s\} \quad \mathbf{1.1}$$

$$V^\pi(s) = E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} / s_t = s\} \quad \mathbf{1.2}$$

Avec $0 \leq \gamma < 1$

Cette fonction est l'espérance (environnement non-déterministe) des renforcements actualisés lorsqu'on part de l'état s ($s_t = s$) et on suit la politique π par la suite.

Si l'environnement est déterministe cette fonction s'écrit comme suit :

$$V^\pi(s) = \{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} / s_t = s \} \quad \mathbf{1.3}$$

Avec $0 \leq \gamma < 1$

De plus cette valeur ne dépend que de l'état s dans le quel on se trouve, car le renforcement dans cet état ne dépend pas des états visités avant d'arriver à l'état s .

4.3.2.2 Fonction « valeur état-action »

Nous allons donner une notre fonction de valeur qui représente la valeur d'un couple « **état-action** » $(s,a) \in S \times A$, pour une politique π . Cette fonction est notée par $Q^\pi(s,a)$, elle est définie de l'ensemble état-action dans les réels, $Q^\pi : S \times A \rightarrow \mathbb{R}$ donc cette fonction se définit comme suit :

Avec $0 \leq \gamma < 1$.

$$Q^\pi(s, a) = E_\pi \{ r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots / s_t = s, a_t = a \} \quad \mathbf{1.4}$$

$$Q^\pi(s, a) = E_\pi \{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} / s_t = s, a_t = a \} \quad \mathbf{1.5}$$

Cette fonction est l'espérance (environnement non déterministe) de la somme des renforcements actualisés lorsqu'on part de l'état s et on exécute l'action a en s ($s_t = s$) et en suivant la politique π à l'instant t .

Si l'environnement est déterministe cette fonction s'écrit comme suit :

$$Q^\pi(s, a) = \{ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} / s_t = s, a_t = a \} \quad \mathbf{1.6}$$

Avec $0 \leq \gamma < 1$

4.3.2.3 Fonction de valeur optimale

La résolution du problème d'apprentissage par renforcement consiste à trouver une politique optimale qui donne un maximum de récompenses à l'agent.

Une politique π est supérieure à une politique π' si : $V^\pi(s) \geq V^{\pi'}(s)$ pour les états possibles et on note cela par $\pi' \leq \pi$. Il existe au moins une politique qui est meilleure de toutes les autres

politique, appelée **politique optimale** qui est noté par π^* , cette politique π^* à une fonction valeur d'état notée par v^* appelée **fonction de valeur d'état optimale** tel que :

$$V^*(s) = \max_{\pi}[V^{\pi}(s)] \quad 1.7$$

Cette politique à aussi une **fonction de valeur d'état-action optimale** notée Q^* est défini par :

$$Q^*(s) = \max_{\pi}[Q^{\pi}(s, a)] \quad 1.8$$

5 Système d'apprentissage par renforcement

Dans le paragraphe précédent nous avons présenté une formalisation du problème d'apprentissage par renforcement. Ceci nous permet maintenant de donner une présentation à ce système d'apprentissage qui est basé sur les éléments suivants :

5.1 Agent

On appelle agent toute entité physique ou virtuelle qui est:

- a- Capable d'agir dans un environnement.
- b- Capable de percevoir son environnement.
- c- Possède des ressources propres.

Ces caractéristiques peuvent être dotées à un agent et il existe d'autres.

5.2 Le temps

L'espace de temps à des formes différentes, il peut être :

- Discret ou continu.
- Fini ou infini.
- Déterministe ou aléatoire.

La plupart des études sur l'apprentissage par renforcement utilisent un espace de temps discret ([Mezaache, 2008](#)).

5.3 Les états

Les états caractérisent les situations de l'agent et de l'environnement à chaque instant, ils peuvent se décomposer en trois formes :

- Une situation relationnelle de l'agent par rapport à l'environnement (position).
- Une situation propre à l'environnement (Modification du milieu).
- Une situation interne à l'agent (sa mémoire, ses capteurs... etc.).

Les trois formes d'état peuvent être présentées en même temps en fonction du problème traité.

5.4 Les actions

L'agent choisit une action parmi les actions possible à chaque instant t , cette action peut être instantanée ou durer jusqu'au prochain instant. A chaque état de l'espace d'état est associé un ensemble d'actions possibles de l'espace d'action, cette relation est représentée sur la figure suivante :

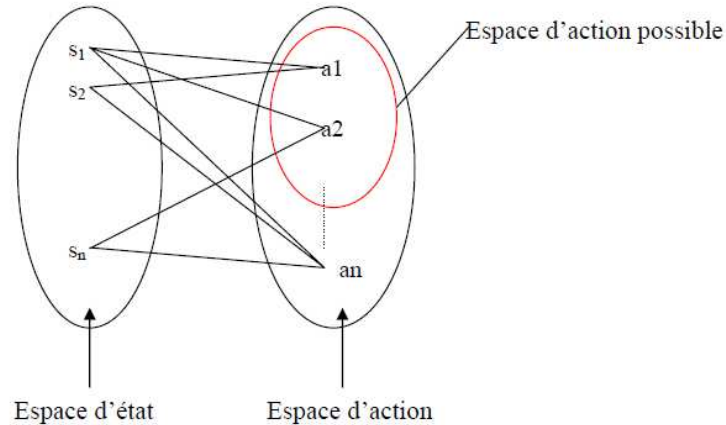


Figure 8 Relation états, actions (Mezaache, 2008)

5.5 Le signal ou la fonction de renforcement

A chaque instant d'interaction, une valeur de renforcement r_t est produite, qui est une valeur numérique bornée qui mesure la justesse de la réaction de l'agent. Le but de l'agent est de maximiser le « Cumul » de ces renforcements dans le temps. Pour prendre en compte

l'horizon de temps, il suffit de considérer la somme des valeurs de renforcement qui sera reçu dans le futur :

$$R_t = r_{t+1} + r_{t+2} + \dots + r_T \quad 1.9$$

Où T est l'instant final qui met fin à l'interaction.

Pour éviter la divergence de on utilisera un facteur de pondération γ , tel que :

$$\gamma \in [0, 1[$$

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \quad 1.10$$

De plus ce facteur de pondération détermine la valeur présente d'une récompense future

5.6 L'environnement

Initialement, l'agent est mis dans un espace de travail dynamique appelé environnement. Le but ou l'objectif du système d'apprentissage par renforcement est d'apprendre une projection topologique des situations « états » vers les actions par interactions *essai* et *erreur* avec cet environnement. Le système d'apprentissage par renforcement doit être capable d'observer parfaitement toutes les informations de l'environnement il est alors possible de choisir une action pertinente basée sur les états réels. Les observations « états ou situations » de l'agent peuvent être les résultats des mesures de capteurs, ainsi les actions peuvent être des tâches à exécuter, un changement de tension ou courant des moteurs,etc.

Il existe plusieurs types d'environnements qui sont utilisés pour les méthodes d'apprentissage; nous citons par exemples :

- **Continu / Discret** : S'il existe un nombre limité de perceptions et d'actions possibles, on parle donc d'un environnement discret.
- **Dynamique / statique** : Un environnement est dit dynamique si son état peut changer en fonction du temps pendant la prise de décision de l'agent. Il est dit statique dans le cas contraire.

- **Déterministe / non déterministe** : On parle d'un environnement déterministe s'il est complètement déterminé par son état courant et par l'action sélectionnée par l'agent.

Donc on peut donner un modèle standard de l'apprentissage par renforcement, on sait qu'il existe des interactions entre l'agent et son environnement généralement sous formes de mesures des capteurs. L'agent doit alors choisir une action de telle manière à améliorer les performances de son comportement. Chaque action doit être jugée par un signal de renforcement. On trouve donc trois parties fondamentales qui réalisent un système d'apprentissage par renforcement qui sont : Environnement, Fonction ou signal de renforcement, Fonction de valeur.

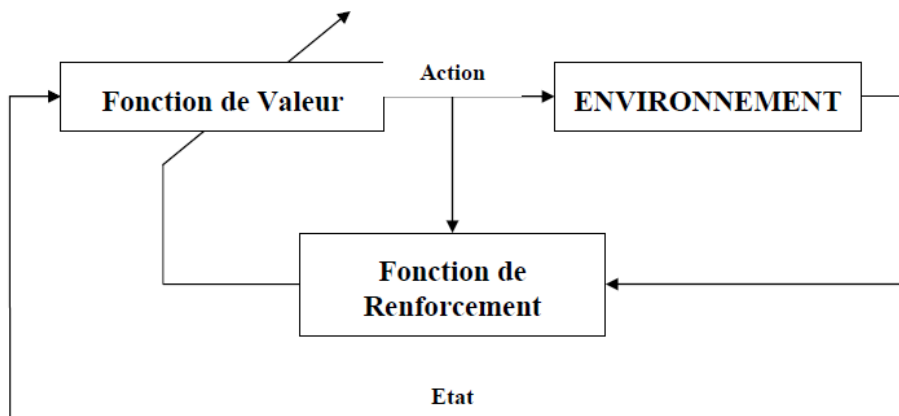


Figure 9 Parties d'un système d'apprentissage par renforcement (Mezaache, 2008)

6 Méthodes de résolution

La résolution du problème d'apprentissage par renforcement est basée sur l'une des méthodes suivantes :

- Méthode de **la programmation dynamique** « PD ».
- Méthode de **Monte Carlo** « MC ».
- Méthode des **différences temporelles** « TD ».

Dans la suite nous intéressons sur la Méthode des **différences temporelles** spécifiquement l'algorithme *Q-Learning*.

6.1 Méthode de la Programmation Dynamique

La programmation dynamique est une méthode de résolution des problèmes d'optimisation. Elle correspond à un ensemble ou une collection d'algorithmes qui sont basés sur la présence d'un modèle parfait de l'environnement pour calculer une politique / stratégie optimale.

Les algorithmes utilisés en *Programmation Dynamique* pour la résolution du problème d'Apprentissage par Renforcement ont une utilisation limitée à cause de :

- Leurs besoins d'un modèle parfait pour l'environnement.
- Ils ont un coût de calcul élevé.

Mais ces algorithmes de la programmation dynamique restent avec une grande utilité et importance théorique car ils sont bien développés mathématiquement.

L'idée principale ou générale de la programmation dynamique est de déterminer une politique optimale, ceci est fait par le calcul des fonctions valeurs optimale V^* ou Q^* en utilisant les équations de Bellman. La recherche d'une politique optimale est basée sur deux phases ou étapes suivantes:

6.1.1 Evaluation d'une politique

La première étape de la programmation dynamique est l'estimation de la fonction valeur pour une politique π donnée. Ceci est fait soit en résolvant le système d'équation de Bellman, tel que :

$$V^\pi(s) = \sum_{a \in A(s)} \pi(s, a) \sum_{s' \in S} T(s, a, s') [R(s, a, s') + \gamma V^\pi(s')] \quad \forall s \in S \quad \mathbf{1.11}$$

Cette équation nous donne un système de s équation linéaire.

Où la fonction valeur d'état est donnée par :

$$V^\pi(s) = E_\pi \{ r_{t+1} + \gamma \sum \gamma^k r_{t+k+2} \quad / \quad s_t = s \} \quad \mathbf{1.12}$$

Soit en utilisant un algorithme d'évaluation itérative d'une politique.

6.1.2 Amélioration d'une politique

L'étape suivante de la programmation dynamique consiste à améliorer la politique π . Après avoir calculé V^π dans l'étape précédente pour un état donné « s » selon la politique π , serait il mieux de choisir une action $a \neq \pi(a)$?

On sait que :

$$(s, a) = E_\pi\{r_{t+1} + \gamma V^\pi(s') \quad / \quad s_t = s, a_t = a\} \quad \mathbf{1.13}$$

$$Q^\pi(s, a) = \sum_{s' \in S} T(s, a, s') [R(s, a, s') + \gamma V^\pi(s')] \quad \mathbf{1.14}$$

Si on a $Q^\pi(s, a) > V^\pi(s)$: il est mieux de basculer vers l'action « a » pour l'état « s », cette opération doit être effectuée pour tout les états, ceci nous conduit d'avoir une nouvelle politique π' au lieu de π .

Avec :

$$\pi(s) = \mathbf{arg \max}_a Q^\pi(s, a)$$

$$\pi(s) = \mathbf{arg \max}_a E_\pi\{r_{t+1} + \gamma V^\pi(s') \quad / \quad s_t = s, a_t = a\} \quad \mathbf{1.15}$$

$$\pi(s) = \mathbf{arg \max}_a \sum_{s' \in S} T(s, a, s') [R(s, a, s') + \gamma V^\pi(s')]$$

Où

\mathbf{argmax} : Désigne l'action qui maximise l'expression $(T(s, a, s') [R(s, a, s') + \gamma V^\pi(s')])$ pour toutes les actions « a » possibles.

Alors :

$$\forall s \in S : Q^\pi(s, \pi(s)) \geq V^\pi(s)$$

Si la nouvelle politique π' égale à π ($\forall s \in S : \pi(s) = \pi'(s)$) alors on atteint la politique optimale.

Donc on a :

$$\forall s \in S : V^\pi(s) = \mathbf{arg \max}_a \sum_{s' \in S} T(s, a, s') [R(s, a, s') + \gamma V^\pi(s')] \quad \mathbf{1.16}$$

Alors on peut donner le schéma suivant qui donne un aperçu sur la méthode suivi pour déterminer une politique optimale suivant la programmation dynamique.

$$\pi_0 \xrightarrow{E} V^{\pi_0} \xrightarrow{A} \pi_1 \xrightarrow{E} V^{\pi_1} \xrightarrow{A} \pi_2 \xrightarrow{E} \dots \dots \dots \xrightarrow{A} \pi^* \xrightarrow{E} V^* \quad 1.17$$

Où :

E : Evaluation d'une politique.

A : Amélioration d'une politique.

6.2 Méthode de Monté Carlo

L'idée de la programmation dynamique généralisée est associée avec l'idée de Monte Carlo (MC). Le terme Monte Carlo consiste à effectuer des simulations, à retenir les résultats des simulations et à calculer des moyennes de résultats. La stratégie du Monte Carlo est adaptée lorsque l'on ne connaît pas le modèle du domaine, c'est une stratégie simple. Il faut que les tâches de l'agent soient décomposées en épisodes, et un épisode correspondra à une simulation. Cette méthode décrit, à partir d'une politique p , comment obtenir une estimation de la fonction de valeur d'états V_p , puis comment estimer une fonction de valeur d'actions Q_p , puis comment améliorer une politique (ce qu'on appelle le « contrôle Monte Carlo »).

La stratégie consiste à lancer des épisodes dans lequel l'agent utilise la politique p . Dans un épisode, pour chaque état, on enregistre le résultat de l'épisode et on dit que $V(s)$ est la moyenne des résultats enregistrés. $V(s)$ converge vers $V_p(s)$.

On démarre d'un état et on effectue une série de transitions conduisant à un état final où le résultat est la somme des récompenses reçues pendant l'épisode. Ce résultat est enregistré dans l'état de départ pour servir au calcul de la moyenne.

$$\text{La règle de mise à jour est : } V(s_t) \leftarrow V(s_t) + \alpha[R_t - V(s_t)]$$

Dans cette méthode, la valeur d'un état est indépendante de la valeur d'un autre état voisin. Par ailleurs on voit que le coût de cette méthode ne dépend que de la longueur des épisodes. Et que l'on peut estimer les valeurs d'états d'un sous-ensemble de l'espace des états, si seul ce sous ensemble nous intéresse, et ignorer complètement les autres états.

Chapitre II: Apprentissage Par Renforcement

Le contrôle Monte Carlo correspond à un processus alternant une amélioration de politique et à une évaluation de politique c'est-à-dire à un exemple d'IGP.

L'algorithme utilisé est constitué des étapes suivantes:

- 1- Générer une trajectoire (une séquence (S, a)) selon la politique π
- 2- Estimer les valeurs des $V^\pi(s_i)$ selon les observations effectuées sur la trajectoire.
- 3- Améliorer la politique π et retourner à l'étape 1.

6.2.1 Discussion

- ✓ Les méthodes Monte Carlo ne présupposent aucune connaissance à priori d'un modèle mais elles ne sont pas incrémentales.
- ✓ Les méthodes de Monte-Carlo utilisent seulement les résultats de l'interaction réelle ou simulée avec un environnement.
- ✓ L'apport principal de ces méthodes réside dans la manière d'estimer la valeur d'un état sur la base de la réception de valeurs successives de récompense cumulée associée à cet état lors de plusieurs trajectoires distinctes.
- ✓ Ces méthodes sont peu utilisées en pratique, car leur fonctionnement exige que l'apprentissage soit décomposé en une succession d'épisodes de longueur finie, faute de quoi la mise à jour de l'estimation de la valeur des états ne peut avoir lieu.
- ✓ Leurs propriétés de convergence ne sont pas encore claires et leur efficacité a été en pratique peu évaluée.
- ✓ Dans ces méthodes, le temps n'est pas une variable explicite.

Nous allons présenter maintenant les méthodes de différence temporelle, qui sont les plus utilisées dans la communauté de l'apprentissage par renforcement.

6.3 Les méthodes différences temporelles (TD)

Si on voudrait identifier une idée centrale et nouvelle de l'apprentissage par renforcement, c'est sans doute la méthode des *Différences Temporelles* qui a été développée par **R. Sutton**.

Les méthodes d'apprentissage par Différences Temporelles sont basées sur l'association des idées des méthodes de la **Programmation Dynamique** et méthodes de **Monté Carlo**.

Les méthodes d'apprentissage par *Différences Temporelles* apprennent à partir de l'expérience directe de l'agent sans avoir un modèle de l'environnement comme les méthodes de **Monté Carlo**.

Pour l'estimation de la valeur d'un état, les méthodes d'apprentissage par Différences temporelles ce fait en fonctions des états voisins sans dépendre des états finals comme les méthodes de la Programmation Dynamique.

Les méthodes Différences Temporelles calculent pendant une séquence les nouvelles évaluations à partir des évaluations précédentes.

Soit « s » un état non terminal visité à l'instant « t » la mise à jour de la fonction, V^π est menée sur la base de ce qui arrive après cette visite (**David, 2004**).

$$V_{t+1}^\pi(S) \leftarrow V^t(S) + \alpha[r_{t+1} + \gamma V_t^\pi(S') - V_t^\pi(S)] \quad 1.18$$

Ainsi l'erreur de prédiction du gain estimé est :

$$r_{t+1} + \gamma V_t^\pi(S') - V_t^\pi(S) \quad 1.19$$

L'algorithme de prédiction des méthodes Différences Temporelles est le suivant :

Initialisation arbitraire de $V^\pi(S)$, π est la politique à évoluer.

Répéter (pour chaque épisode)

Initialiser s

Répéter pour chaque itération d'épisode

$a \leftarrow$ Action donné par π pour l'état s

Exécuter l'action a, Observer la récompense r et l'état suivant s'

$$V(s) \leftarrow V(s) + \alpha[r_{t+1} + \gamma V(s') - V(s)]$$

$$s \leftarrow s'$$

Jusqu'à l'état terminal s.

Chapitre II: Apprentissage Par Renforcement

Tableau 1 Algorithme de prédiction des méthodes Différences Temporelles (Mezaache, 2008)

Dans les méthodes de Différence Temporelles l'évaluation se fait à un pas de temps et elles n'ont pas besoin d'un modèle de l'environnement. Plusieurs algorithmes sont liés à ces méthodes par exemple : SARSA, TD(λ), R-Learning, Q-Learning.....

Dans le paragraphe suivant nous allons donner une bonne description de l'algorithme

Q-Learning

6.3.1 Q-Learning

Définition : L'algorithme Q-Learning se classe dans les méthodes de Différence Temporelles comme une méthode de résolution du problème d'apprentissage par renforcement pour un modèle d'environnement inconnu, c'est une méthode qui se base sur l'équation réactualisation :

$$Q_{t+1}(s, a) = Q(s, a) + \alpha[r_{t+1} + \gamma \max_{a'} Q_t(s', a') - Q_t(s, a)] \quad \mathbf{1.20}$$

De plus cette méthode est basée sur les fonctions suivantes :

- **Une fonction de sélection**

A partir de l'état actuel qui est observée par l'agent, une action est choisie et exécutée en se basant sur la connaissance disponible au sein de la mémoire interne (Cette connaissance est stockée sous forme de valeur d'utilité associée à une paire « Etat, Action »).

- **Une fonction de renforcement**

Après l'exécution de l'action dans le monde réel. La fonction de renforcement utilise la nouvelle situation pour générer la valeur de renforcement. Ce renforcement prend en général une simple valeur +1, -1 ou 0.

- **Une fonction de mise à jour**

Cette fonction utilise la valeur de renforcement pour ajuster la valeur associée à l'état ou bien à la paire « Etat, Action » qui vient d'être exécutée.

Le *Q-Learning* a comme principe l'estimation de la fonction Q^* qui est définie par :

$$Q^*(s,a) = E[r_{t+1} + \gamma V^*(s)] = E[r_{t+1} + \gamma \max_{a'} Q^*(s', a')] \quad 1.21$$

En utilisant l'équation de réactualisation on trouve que :

$$Q_{t+1}^\pi(s_t, a_t) = (1 - \alpha_t) Q_t^\pi(s_t, a_t) + \alpha_t [r_{t+1} + \gamma \max_{a'} (Q_t^\pi(s', a'))] \quad 1.22$$

Où :

r_{t+1} est le renforcement reçu quand on choisie l'action « a » dans l'état « s » pour passer à l'état « s' ».

α_t est un réel positif : $\alpha_t \in]0,1[$.

En principe il faut *explorer* aléatoirement l'environnement pendant un grand nombre d'itérations pour que le *Q-Learning* converge vers la fonction Q optimale, en suite on peut utiliser la politique optimale définie par :

$$\pi'(s) = \arg \max_{a \in A} Q^*(s, a) \quad 1.23$$

• Politique d'Exploration / Exploitation

La politique optimale est obtenue en choisissant l'action qui, à chaque état maximise la fonction de valeur Q . Pour avoir une bonne estimation de cette fonction, on doit balayer et évoluer l'ensemble des actions possibles, pour tous les états. C'est ce que l'on appelle la phase d'exploration par rapport à l'exploitation ([Glo, 2015](#)).

On note *PEE(s)* la Politique d'Exploration / Exploitation qui permet le choix d'une action suivant cette politique dans l'état « s ».

L'algorithme de *Q Learning* s'écrit comme suit

Initialisation de $Q(s,a)$ arbitrairement

Pour chaque épisode

Choisir un état de départ «s»

Pour chaque itération dans l'épisode

Choisir «a» en fonction $Q(s,a)$ selon la stratégie d'exploration exécuter a.

Observer s et r

$$Q_{t+1}^\pi(s_t, a_t) = (1 - \alpha_t) Q_t^\pi(s_t, a_t) + \alpha_t [r_{t+1} + \gamma \max_{a'} (Q_t^\pi(s', a'))]$$

$s \leftarrow s'$

Jusqu'à l'état final «s».

Chapitre II: Apprentissage Par Renforcement

Tableau 2 Algorithms Q-Learning (Mezaache, 2008)

7 Discussion

Nous terminons cette partie en dressant un tableau comparatif des méthodes d'apprentissage par renforcement précédemment citées:

| | Méthodes Programmation dynamique | Méthodes Monté Carlo | Méthodes Différence Temporelle |
|--|--|--|--|
| Il faut connaître le modèle du processus | Oui ($\pi(s,a), P, R,$) à connaître | Non | Non |
| Le calcul implique une valeur estimée | Oui | Non | Oui |
| Profondeur de l'espace de recherche | Scrute toutes les actions possibles à partir d'un état et actualise à partir des valeurs estimées des états suivants | Déroule un épisode Complet pour apprendre | Actualise à partir de du renforcement immédiat et de la valeur estimée de l'état suivant |
| Preuves de convergence | Oui | Convergence vers π^* non prouvée | Oui |
| Complexité de calcul | Temps de calcul polynomial avec le nombre d'états | Estimation plus précise si grand nombre d'épisodes. | Calcul simple, sur tous les états un grand nombre de fois |
| Avantages | Théoriquement important Planification (optimisation, commande optimale) | Basé seulement sur les résultats de l'interaction réelle ou simulée avec un environnement. Pas besoin de scruter tous les états. | Apprentissage sans attendre la fin de l'épisode. Conduit à de nombreux algorithmes d'apprentissages. |
| Inconvénients | Limité à des problèmes de quelques millions d'états. | Efficacité en pratique peu évaluée Seulement pour des tâches épisodiques | Limitation d'implémentation Avec des variables continue. |

Tableau 3 Comparaison des méthodes d'apprentissage par renforcement

8 L'apprentissage par renforcement en profondeur

8.1 Qu'est-ce que l'apprentissage en profondeur

Ils ne font que rechercher dans un ensemble prédéfini d'opérations, appelé espace d'hypothèses.

Techniquement, c'est l'apprentissage automatique: rechercher des représentations utiles de certaines données d'entrée, dans un espace de possibilités prédéfini, en s'appuyant sur un signal de retour. Cette idée simple permet de résoudre un très large éventail de tâches intellectuelles, de la reconnaissance de la parole à la conduite autonome.

Maintenant que vous comprenez ce que nous entendons par apprentissage, examinons ce qui rend l'apprentissage en profondeur spécial.

8.2 Le «profond» dans l'apprentissage en profondeur

L'apprentissage en profondeur est un sous-domaine spécifique de l'apprentissage automatique: une nouvelle approche de l'apprentissage des représentations à partir de données qui met l'accent sur l'apprentissage de couches successives de représentations de plus en plus significatives. L'apprentissage en profondeur n'est pas une référence à une forme de compréhension plus profonde obtenue par l'approche; il représente plutôt cette idée de couches successives de représentations. Le nombre de couches contribuant à un modèle de données est appelé profondeur du modèle. D'autres noms appropriés pour le champ auraient pu être apprentissage des représentations en couches et apprentissage des représentations hiérarchiques. L'apprentissage en profondeur moderne implique souvent des dizaines, voire des centaines, de couches successives de représentations - et elles s'apprennent toutes automatiquement à partir de l'exposition aux données d'entraînement. Dans le même temps, d'autres approches de l'apprentissage automatique ont tendance à privilégier l'apprentissage d'une ou deux couches de représentations des données; Par conséquent, ils sont parfois appelés apprentissage superficiel.

Chapitre II: Apprentissage Par Renforcement

En apprentissage en profondeur, ces représentations en couches sont apprises (presque toujours) au moyen de modèles appelés réseaux de neurones, structurés en couches littérales superposées. Le terme réseau de neurones est une référence à la neurobiologie, mais bien que certains des concepts centraux de l'apprentissage en profondeur aient été développés en partie en s'inspirant de notre compréhension du cerveau, les modèles d'apprentissage en profondeur ne sont pas des modèles du cerveau. Rien ne prouve que le cerveau mette en œuvre des mécanismes d'apprentissage utilisés dans les modèles modernes d'apprentissage en profondeur. Vous pouvez rencontrer des articles de science pop affirmant que l'apprentissage en profondeur fonctionne comme le cerveau ou a été modélisé sur le cerveau, mais ce n'est pas le cas. Ce serait déroutant et contre-productif pour les nouveaux arrivants sur le terrain de penser que l'apprentissage en profondeur est en aucune manière lié à la neurobiologie; vous n'avez pas besoin de ce voile de mystique et de mystère «comme notre esprit», et vous pouvez aussi oublier tout ce que vous avez pu lire sur les liens hypothétiques entre l'apprentissage en profondeur et la biologie. Pour nos besoins, l'apprentissage en profondeur est un cadre mathématique pour l'apprentissage des représentations à partir de données.

À quoi ressemblent les représentations apprises par un algorithme d'apprentissage en profondeur? Voyons maintenant comment un réseau de plusieurs couches (voir figure 10) transforme l'image d'un chiffre afin de reconnaître son chiffre.

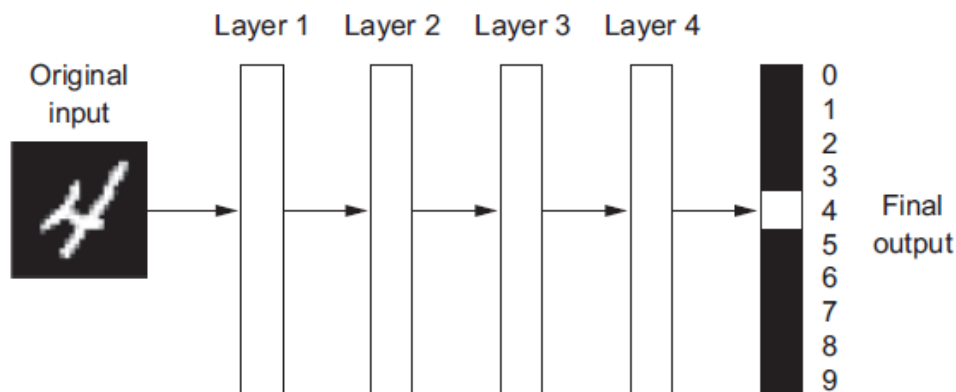


Figure 10 Un réseau neuronal profond pour la classification des chiffres (François, 2018)

Comme vous pouvez le (voir à la figure 11), le réseau transforme l'image numérique en représentations de plus en plus différentes de l'image d'origine et de plus en plus informatives sur le résultat final. Vous pouvez considérer un réseau en profondeur comme une opération de

distillation d'informations en plusieurs étapes, où l'information passe par des filtres successifs et ressort de plus en plus épurée (c'est-à-dire utile pour certaines tâches).

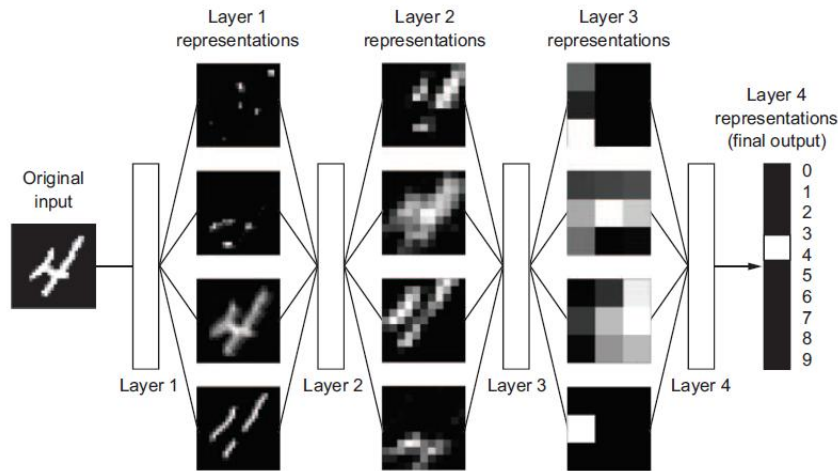


Figure 11 Représentations profondes apprises par un modèle de classification numérique (François, 2018)

Donc, c'est ce que l'apprentissage en profondeur est techniquement: une méthode à plusieurs étapes pour apprendre les représentations de données. C'est une idée simple, mais il se trouve que des mécanismes très simples, suffisamment dimensionnés, peuvent ressembler à de la magie.

8.3 Comprendre le fonctionnement de l'apprentissage en profondeur, en trois chiffres

À ce stade, vous savez que l'apprentissage automatique consiste à mapper des entrées (telles que des images) sur des cibles (telles que l'étiquette «chat»), en observant de nombreux exemples d'entrées et de cibles. Vous savez également que les réseaux de neurones profonds font cette entrée-à-cible

Mappage via une séquence profonde de transformations de données simples (couches) et que ces transformations de données sont apprises par exposition à des exemples. Voyons maintenant comment cet apprentissage se déroule concrètement.

La spécification de ce que fait une couche sur ses données d'entrée est stockée dans les pondérations de la couche, qui sont essentiellement des nombres. En termes techniques, nous dirions que la transformation mise en œuvre par une couche est paramétrée par ses

Chapitre II: Apprentissage Par Renforcement

pondérations (voir figure 12). (Les poids sont aussi parfois appelés paramètres d'une couche.) Dans ce contexte, l'apprentissage signifie rechercher un ensemble de valeurs pour les pondérations de toutes les couches d'un réseau, de telle sorte que le réseau mapperait correctement les exemples d'entrées sur leurs cibles associées. Mais voici la chose: un réseau de neurones profonds peut contenir des dizaines de millions de paramètres. Trouver la valeur correcte pour chacun d'eux peut sembler une tâche ardue, d'autant plus que la modification de la valeur d'un paramètre affectera le comportement de tous les autres!

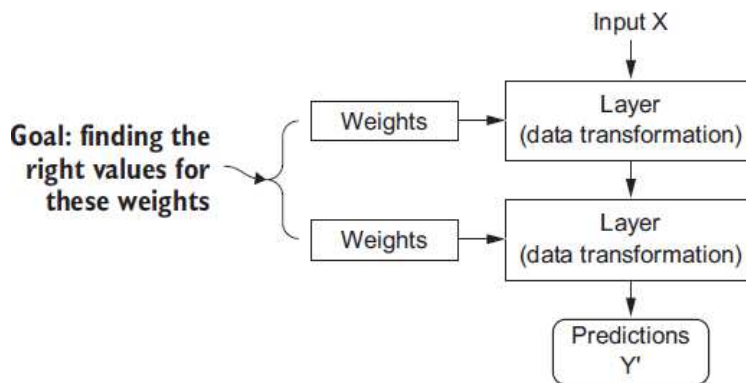


Figure 12 Un réseau de neurones est paramétré par ses poids (François, 2018)

Pour contrôler quelque chose, vous devez d'abord pouvoir l'observer. Pour contrôler la sortie d'un réseau de neurones, vous devez pouvoir mesurer dans quelle mesure cette sortie est différente de celle que vous attendiez. C'est le travail de la fonction de perte du réseau, également appelée fonction objectif. La fonction de perte prend les prédictions du réseau et de la vraie cible (ce que vous vouliez que le réseau produise) et calcule un score de distance, rendant compte de la performance du réseau pour cet exemple spécifique (voir figure 13).

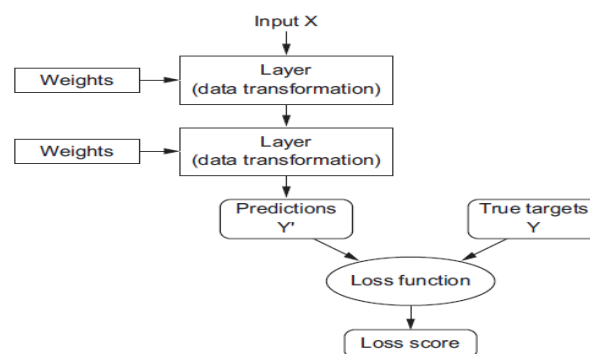


Figure 13 Une fonction de perte mesure la qualité de la sortie du réseau (François, 2018)

L'astuce fondamentale de l'apprentissage en profondeur consiste à utiliser ce score comme signal de rétroaction pour ajuster légèrement la valeur des poids, dans un sens qui abaissera le score de perte pour l'exemple actuel (voir la figure 14). Cet ajustement est le travail de l'optimiseur, qui met en œuvre ce que l'on appelle l'algorithme de rétro propagation: l'algorithme central de l'apprentissage en profondeur.

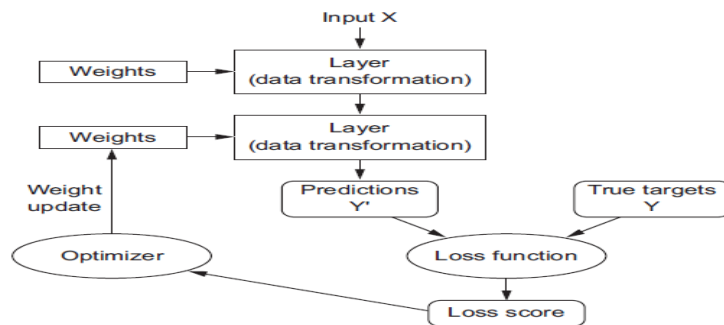


Figure 14 Le score de perte sert de signal de retour pour ajuster les poids (François, 2018)

Initialement, des valeurs aléatoires sont attribuées aux poids du réseau, de sorte que le réseau implémente simplement une série de transformations aléatoires. Naturellement, sa production est loin de ce qu'elle devrait être dans l'idéal et le score de perte est donc très élevé. Mais avec chaque exemple de processus réseau, les poids sont légèrement ajustés dans la bonne direction et le score de perte diminue. Il s'agit de la boucle d'entraînement qui, répétée un nombre suffisant de fois (généralement des dizaines d'itérations sur des milliers d'exemples), donne des valeurs de poids minimisant la fonction de perte. Un réseau avec une perte minimale est un réseau dont les sorties sont aussi proches que possible des cibles: un réseau formé. Encore une fois, il s'agit d'un mécanisme simple qui, une fois mit à l'échelle, finit par sembler.

9 Algorithme de rétro propagation du gradient

9.1 Principe

Chapitre II: Apprentissage Par Renforcement

La rétro propagation du gradient consiste à propager « à l'envers » (de la couche de sortie vers la couche d'entrée) l'erreur obtenue sur les exemples de la base d'apprentissage. On utilise pour cela l'erreur quadratique « le carré de la différence entre ce qu'on obtient et ce qu'on désire ».

Si on calcule la dérivée partielle de l'erreur quadratique par rapport aux poids des connexions (gradient), il est possible de déterminer la contribution des poids à l'erreur générale, et de corriger ces poids de manière à se rapprocher du résultat souhaité. La correction par itération en corrige plus ou moins fortement les poids par l'intermédiaire d'un coefficient (**Châtelain, 2013**).

Après un certain nombre d'itérations, où on n'est satisfait du classement des exemples de notre base d'apprentissage, on fixe les poids qui constituent aussi des frontières entre les classes.

9.2 Algorithme

A-Définition du réseau

Soit un réseau multicouche définit par:

- Une couche d'entrée à m cellules d'entrées. Ces cellules ne sont pas des neurones mais simplement des entrées $\mathbf{x}_i = \mathbf{e}_i$ du réseau.
- Une couche cachée qui contient n neurones qui ont une fonction d'activation \mathbf{y}_i
- Une couche de sortie à p neurones qui ont une fonction d'activation. \mathbf{z}_k
- $n \times m$ connexions entre la couche d'entrée et la couche cachée, chacune pondérée par v_{ji} .
- $p \times n$ connexions entre la couche cachée et la couche de sortie, chacune pondérée par w_{kj}

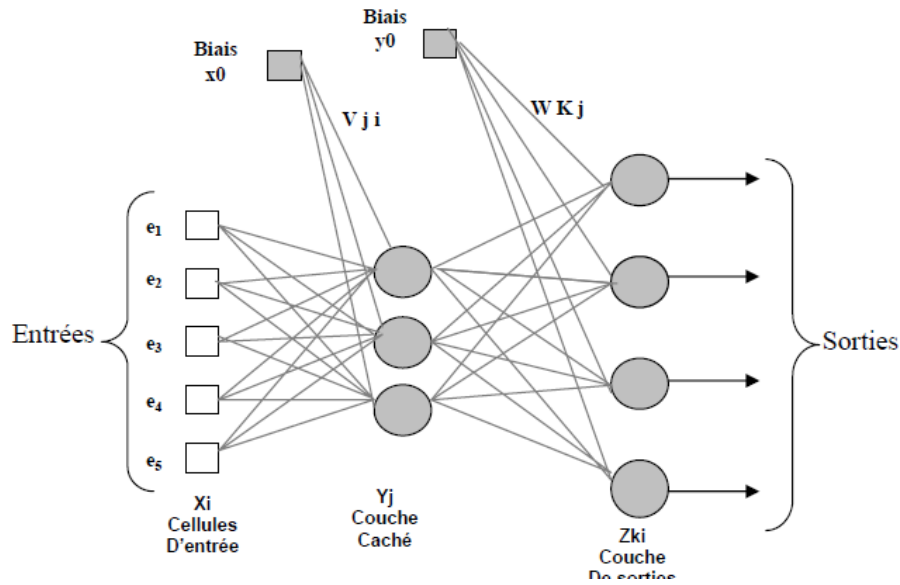


Figure 15 Exemple de réseau MLP à une couche cachée avec 5 entrées, 3 neurones dans la couche cachée, et quatre 4 sorties (Mezaache, 2008)

B- Les Etapes de l'algorithme

❖ **ETAPE 1 : Initialisation au hasard ou aléatoire des poids des connexions**

v_{ij} Et w_{ig}

❖ □ **ETAPE 2 : Propagation des entrées $x_i=e_i$**

○ On propage vers la couche cachée :

$$y_j = f\left(\sum_{i=1}^m x_i v_{ij} + x_0\right) \quad \mathbf{1.24}$$

○ Puis de la couche cachée vers la couche de sortie :

$$z_k = f\left(\sum_{j=1}^n y_j w_{kj} + y_0\right) \quad \mathbf{1.25}$$

Les valeurs x_0 et y_0 sont des Biais, est la fonction d'activation qu'on a choisie où on a définit notre réseau MLP.

Chapitre II: Apprentissage Par Renforcement

- ❖ **ETAPE 3 : rétro propagation de l'erreur** pour chaque neurone de la couche de sortie On calcule l'erreur, c'est-à-dire la différence entre la sortie désirée s_k et la sortie réelle (obtenue) z_k .

$$E_k = z_k(1 - z_k)(s_k - z_k) \quad 1.26$$

On propage cette erreur sur la couche cachée ; l'erreur de chaque neurone de la couche cachée est donnée par :

$$F_j = y_j(1 - y_j) \sum_{k=1}^p w_{kj} E_k \quad 1.27$$

- ❖ **ETAPE 4 : Correction des poids des connexions** Il reste maintenant la modification des poids des connexions et aussi les biais.

- Entre la couche d'entrée et la couche cachée :

$$\begin{cases} \Delta v_{ji} = \eta y_j F_j \\ \Delta x_0 = \eta F_j \end{cases} \quad 1.28$$

- Entre la couche cachée et la couche de sortie :

$$\begin{cases} \Delta w_{kj} = \eta z_k E_k \\ \Delta y_0 = \eta E_k \end{cases} \quad 1.29$$

η Etant un paramètre qu'il reste à déterminer.

- ❖ **BOUCLER à l'étape 2** jusqu'à un critère d'arrêt à définir.

C- Sommaire de l'algorithme Rétro propagation de l'erreur

1. *Initialisation au hasard ou aléatoire des poids des connexions entre les couches (Entrée - Cachée) et (Cachée - Sortie).*
2. *Propagation des entrées $x_i = e_i$ de la couche d'entrée vers la couche de sortie en passant par la couche cachée.*
3. *rétropropagation de l'erreur des neurones de la couche de sortie vers la couche cachée puis vers la couche d'entrée.*
4. *Correction des poids des connexions entre la couche d'entrée et la couche cachée ; la couche cachée et la couche de sortie.*
5. *Boucler à 2 jusqu'à un critère d'arrêt à définir.*

Tableau 4 Sommaire de l'algorithme Rétro propagation de l'erreur(Vincent et al., 2018)

10 L'apprentissage par renforcement en profondeur

L'apprentissage par renforcement en profondeur est la combinaison de l'apprentissage par renforcement (RL) et de l'apprentissage en profondeur. Ce domaine de recherche a été en mesure de résoudre un large éventail de tâches de prise de décision complexes qui étaient auparavant inaccessibles pour une machine. Ainsi, Deep RL ouvre de nombreuses nouvelles applications dans des domaines tels que la santé, la robotique, les réseaux intelligents, la finance, etc. Ce manuscrit fournit une introduction aux modèles, algorithmes et techniques d'apprentissage par renforcement en profondeur. L'accent est mis en particulier sur les aspects liés à la généralisation et sur la profondeur d'utilisation de RL pour des applications pratiques.

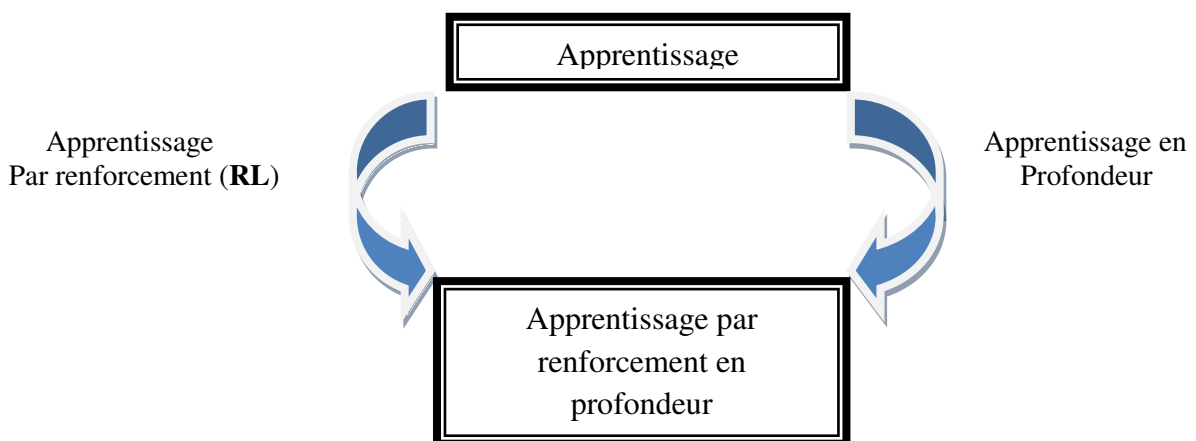


Figure 16 Apprentissage par renforcement en profondeur

Conclusion

La prise de décision séquentielle reste un domaine de recherche actif et de nombreux défis théoriques, méthodologiques et expérimentaux restent à relever. Les développements importants dans le domaine de l'apprentissage en profondeur ont contribué à de nombreuses nouvelles voies dans lesquelles les méthodes RL et l'apprentissage en profondeur sont combinées. En particulier, l'apprentissage en profondeur a apporté d'importantes capacités de généralisation, ce qui ouvre de nouvelles possibilités de travailler avec de grands espaces d'état et / ou d'action de grande dimension. Il y a tout lieu de penser que ce développement se poursuivra dans les années à venir avec des algorithmes plus efficaces et de nombreuses nouvelles applications.

Dans ce chapitre nous avons essayé de donner une description générale pour le problème d'apprentissage par renforcement basée sur la transition d'un état « s » à un état « s' » en exécutant une action « a » qui se fait suivant une politique π et qui est caractérisée par ces fonctions valeurs V et. Un signal de renforcement « r » juge l'action « a » exécutée par l'agent qui doit maximiser ce signal. Les méthodes utilisées pour la résolution de ce problème ont été présentées on a parlé sur la méthode ***Q Learning*** et son algorithme.

CHAPITRE III :

conception

Introduction

Comme nous l'avons présenté dans le chapitre deux, l'apprentissage par renforcement est basé sur des fonctions ou des blocs principaux qui sont, l'environnement, fonction de renforcement et la fonction valeur et on a vu les différentes méthodes utilisées qui sont Monte Carlo, la programmation dynamique et la différence temporelle.

Dans ce chapitre on va essayer de donner une description pour chaque fonction qui participe à la conception du système d'apprentissage par renforcement pour la résolution du problème consommation d'énergie qui ressemble à celui du blackjack. On dressera l'architecture générale de la solution proposée et nous présenterons à la fin la modélisation UML du système.

1 Architecture générale proposé de la solution

Dans un smart home il existe des objets qui sont connectés à internet, ce qui permet les échanges de données, ces objets peuvent prendre la forme de n'importe quel objet qui se trouve dans le smart home, ces objets sont capable de communiquer et transmettre des informations avec d'autres objets connectés.

Ces objets sont équipés de capteurs qui vont collecter des données depuis l'extérieur. Ces données sont soumises à un agent pour l'analyse et l'extraction des connaissances, afin que l'agent sache gérer l'environnement la prochaine fois. Ensuite l'agent intelligent va prendre les décisions. Ces décisions vont être envoyées aux actionneurs qui vont les exécuter.

Utiliser l'apprentissage par renforcement pour l'optimisation d'énergie pour les objets connecté c'est notre objectif .l'apprentissage peut être centré selon quatre axes : agent, environnement, interaction ou organisation (**Vercouter et al., 1998**).Pour l'architecture proposée pour la solution(**Figure 17**), l'agent exécute l'un des algorithmes d'apprentissage par renforcement sur un environnement domicile, avec les états : **demande refroidissement, prix d'électricité**, et avec une récompense : **consommation d'électricité (prix)**.

Environnement : Domicile

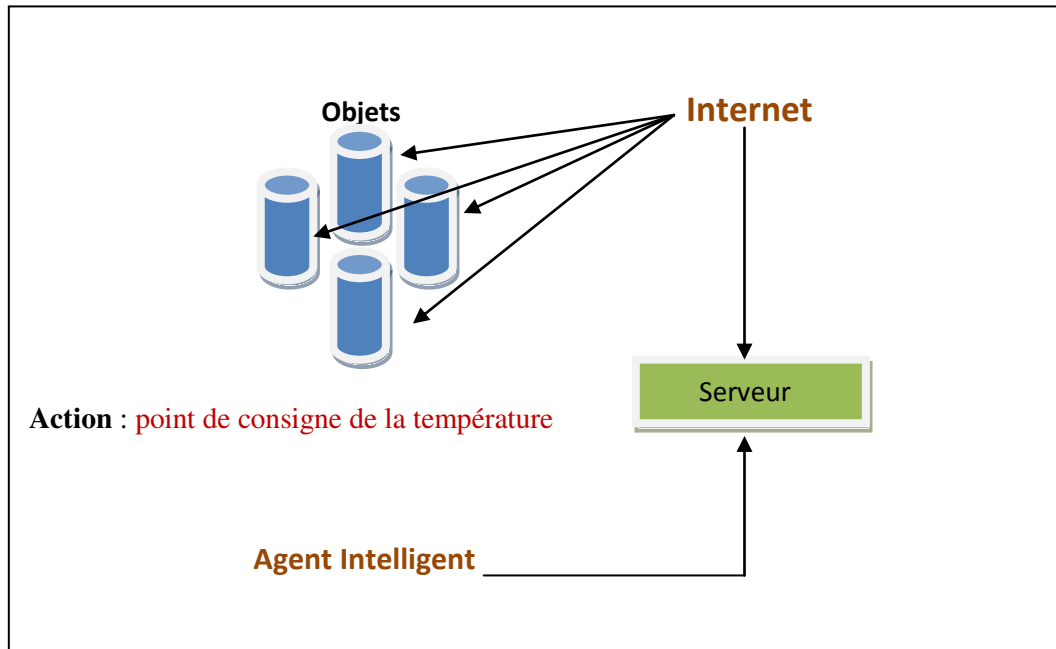


Figure 17 Architecture générale

2 Modélisation de l'Agent Intelligent

Depuis l'origine de l'informatique, un programme est défini comme un ensemble de commandes effectuant un traitement spécifique sur un ensemble de données déterminées. Il est déclenché par un événement, une demande de l'utilisateur par exemple, et fournit un résultat en fin d'exécution. La définition a évolué pour accéder aujourd'hui à la notion d'agents « intelligents ».

Notre problème ressemble à celui du blackjack qui est obtenir des cartes dont la somme des valeurs numériques est aussi grande que possible sans dépasser 21.

Chapitre III : conception

2.1 Concept du Blackjack

Après avoir reçu deux cartes, le joueur tire des cartes pour s'approcher de la valeur 21 sans la dépasser. Le but du joueur est de battre le croupier en obtenant un total de points supérieur à celui-ci ou en voyant se dernière dépasser 21. chaque joueur joue contre le croupier, qui représente la banque, ou le casino, et non contre les autres joueurs.

- Chaque carte numérotée de 2 à 10 a sa valeur nominale (égale au numéro sur la carte)
- Les valets, les dames et les rois (les figures), ont une valeur de 10 points
- L'AS vaut 1 point ou 11 points, au choix du joueur.

Le jeu commence avec deux cartes distribuées au donneur et au joueur. Une des cartes du donneur est face visible et l'autre est face cachée. Si le joueur a 21 immédiatement (un as et une carte de 10), cela s'appelle un naturel. Il gagne ensuite à moins que le croupier ait aussi un naturel, auquel cas le jeu est un match nul. Si le joueur n'a pas de naturel, il peut alors demander des cartes supplémentaires, une par une (touche), jusqu'à ce qu'il arrête (colle) ou dépasse 21 (passe). S'il fait faillite, il perd; s'il s'en tient, alors c'est le tour du croupier. Le croupier frappe ou colle selon une stratégie fixe sans choix: il colle sur toute somme supérieure ou égale à 17, et frappe autrement. Si le croupier fait faillite, le joueur gagne; sinon, le résultat - gagner, perdre ou tirer - est déterminé par la somme finale la plus proche de 21.

2.2 Transposition à notre problème

Dans notre cas on a distingué trois catégories d'appareils qui sont : lampe (appareils à faible consommation énergétique), tv (appareil à moyenne consommation énergétique), réfrigérateur (appareil à fort consommation énergétique). Chaque appareil consomme d'énergies intervalles donne la sommes totales de toutes les énergies ne dépasse pas un certain seuil fixé par l'utilisateur (on utilisera la valeur **seuil**= 10000watts).

- Chaque lampe consomme une valeur d'énergie comprise entre (15,100watts)
- Les TV consomment entre (200,400watts)
- Les réfrigérateurs consomment entre (1000,2000watts)

Chapitre III : conception

L'objectif ici, c'est qu'à chaque étape, un utilisateur humain (l'adversaire) allumera de façon aléatoire des appareils de type lampe, tv et réfrigérateur. Le rôle de l'agent à chaque étape (état), est d'essayer de garder les appareils allumés ou éteindre le minimum possible afin de ne pas dépasser un seuil critique **Seuil**.

Nous commençons par examiner les méthodes de Monte Carlo pour apprendre la fonction de valeur d'état une politique représente la méthode **sample_policy**(observation) qui retourne 0 si le score(consommation énergétique) est supérieur strictement au **seuil** sinon retourne 1.

Rappelez-vous que la valeur d'un état est le rendement attendu (récompense actualisée cumulée attendue) à partir de cet état. Un moyen évident de l'estimer à partir de l'expérience consiste donc simplement à moyenner les rendements observés après des visites dans cet État. Plus on observe de rendements, plus la moyenne devrait converger vers la valeur attendue. Cette idée sous-tend toutes les méthodes de Monte Carlo.

En particulier, notez que la charge de calcul liée à l'estimation de la valeur d'un seul état est indépendante du nombre d'états Cela peut rendre les méthodes de Monte Carlo particulièrement attrayantes lorsque l'on exige la valeur d'un seul ou d'un sous-ensemble d'états. On peut générer de nombreux échantillons d'épisodes à partir des états d'intérêt, en faisant la moyenne des retours de ces états, en ignorant tous les autres.

Notre agent intelligent va extraire les connaissances des informations envoyées par le serveur, à savoir ;

- Le nombre d'appareils de type lampe allumés ainsi que leur consommation énergétique
- Le nombre d'appareils de type TV allumés ainsi que leur consommation énergétique
- Le nombre d'appareils de type lampe réfrigérateurs ainsi que leur consommation énergétique

Ensuite, l'agent va maximiser ses récompenses en utilisant les algorithmes d'apprentissage par renforcement qui sont : monte Carlo, programmation dynamique, différences temporelle pour résoudre le problème d'optimisation, puis prendre des décisions en fin pour exécuter les actions citées déjà qui sont ensuite envoyer au serveur (Figure 18).

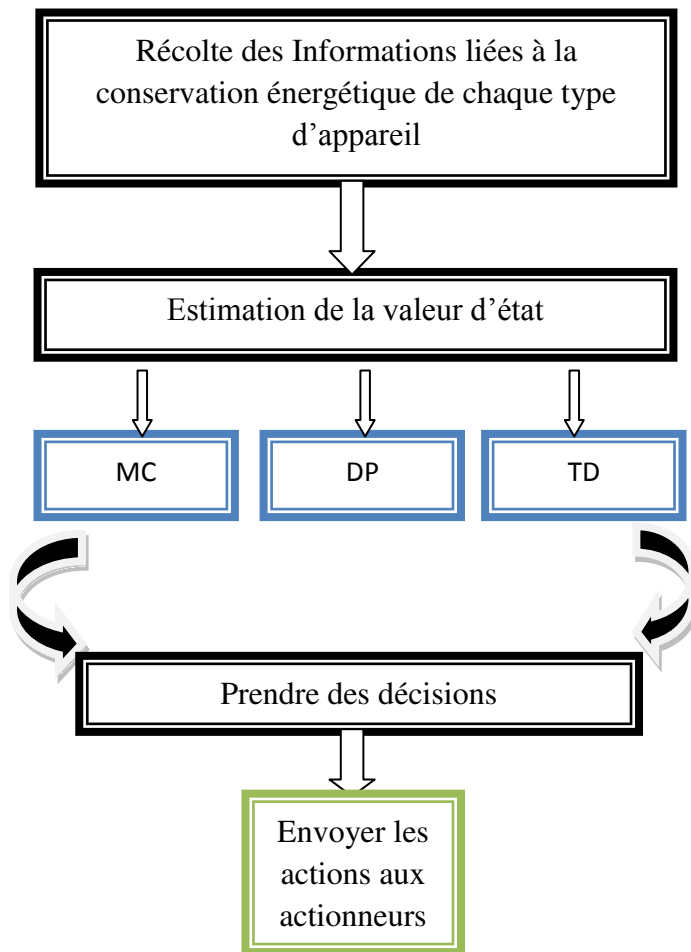


Figure 18 Fonctionnement de Notre Agent

3 Serveur

En informatique, un serveur est un programme qui fournit des services à d'autres programmes (et à leurs utilisateurs) installés sur le même ordinateur ou sur des ordinateurs distincts.

Dans le modèle de programmation client/serveur, un serveur est un programme qui attend et satisfait les demandes émanant de programmes clients installés sur le même ordinateur ou des ordinateurs distincts.

Notre serveur va subir les informations du capteur qui sont ensuite envoyées vers l'agent intelligent, après l'extraction des connaissances et les prises des décisions par l'agent intelligent, les smart homes vont récupérer les données et les actions envoyer par le serveur pour ensuite le système fonction correctement .

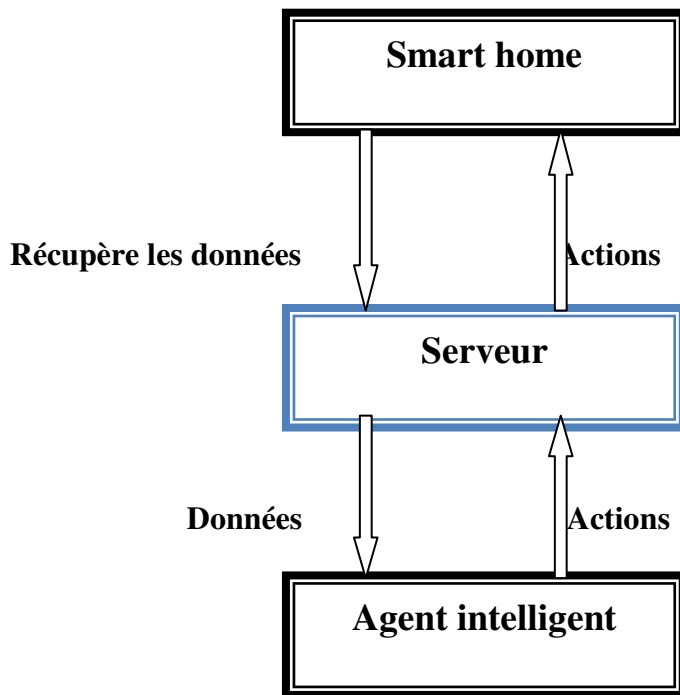


Figure 19 Fonctionnement du Serveur

4 Modélisation UML

La modélisation consiste à créer une représentation simplifiée d'un problème: **le modèle**. Grâce au modèle il est possible de représenter simplement un problème, un concept et le simuler. La modélisation comporte deux composantes :

- L'analyse, c'est-à-dire l'étude du problème
- la conception, soit la mise au point d'une solution au problème Le modèle constitue ainsi une représentation possible du système pour un point de vue donné.

Chapitre III : conception

Le méta modèle UML fournit une panoplie d'outils permettant de représenter l'ensemble des éléments du monde objet (classes, objets, ...) ainsi que les liens qui les relie.

Toutefois, étant donné qu'une seule représentation est trop subjective, UML fournit un moyen astucieux permettant de représenter diverses projections d'une même représentation grâce aux **vues**.

Une vue est constituée d'un ou plusieurs **diagrammes**. On s'intéresse sur le **vues dynamique** qui montre le fonctionnement du système.

Dans ce mémoire deux diagrammes sont utilisés diagramme de **séquence** et diagramme **d'états-transitions**.

4.1 Diagramme de séquence

Un diagramme de séquence est un diagramme UML (Unified Modeling Language) qui représente la séquence de messages entre les objets au cours d'une interaction. Un diagramme de séquence comprend un groupe d'objets, représentés par des lignes de vie, et les messages que ces objets échangent lors de l'interaction.

Les diagrammes de séquence représentent la séquence de messages transmis entre des objets. Ils peuvent également représenter les structures de contrôle entre des objets.

Un message est un élément de diagramme Unified Modeling Language (UML) qui définit un type particulier de communication entre les instances au cours d'une interaction. Un message fait circuler des informations d'une instance, représentée par une ligne de vie, à une autre instance au cours d'une interaction.

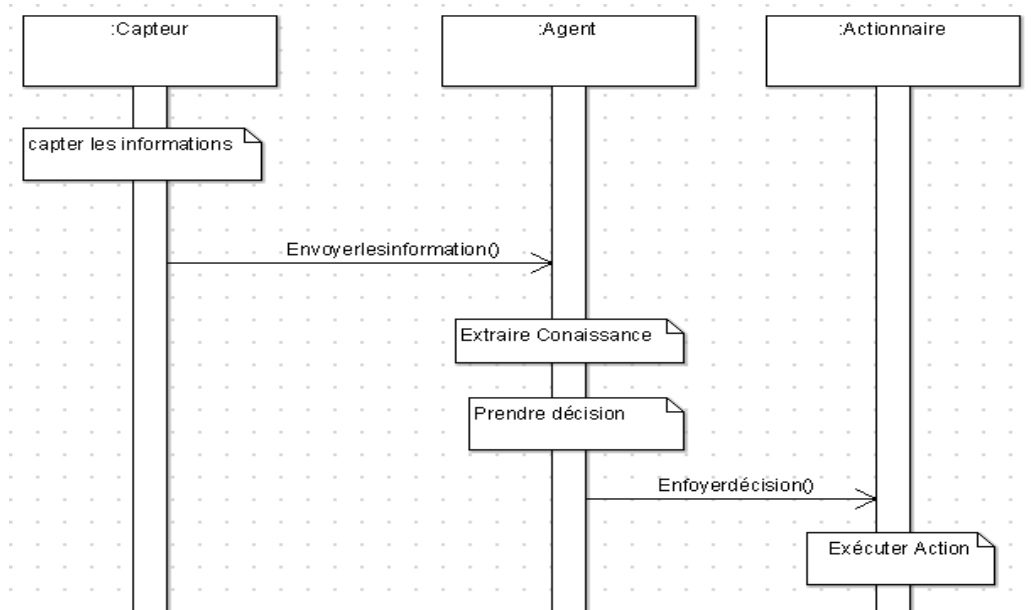


Figure 20 diagramme de séquence

Le diagramme de séquence permet de montrer comment les objets interagissent et communiquent entre eux pour accomplir une tâche du système. La variable indépendante est ici le temps.

Notre diagramme débute la tâche par le capteur qui capte les informations puis l'envoyer vers l'agent qui va extraire les connaissances et prendre des décisions qui sont envoyés vers l'actionnaire. L'actionnaire va exécuter l'action.

4.2 Diagrammes d'états-transitions

Un diagramme d'états-transitions est un diagramme UML qui fournit une représentation graphique d'une State Machine, le comportement public d'un classificateur (composant ou classe), sous la forme des changements de l'état du classificateur et des événements qui permettent la transition d'un état à l'autre.

Chapitre III : conception

A la différence des autres diagrammes d'interaction, le diagramme d'états-transitions peut représenter une spécification complète des scénarios possibles appartenant au classificateur. A un moment donné, l'objet doit être dans l'un des états définis.

Les diagrammes d'états-transitions montrent le comportement du classificateur via des règles d'exécution qui expliquent précisément de quelle façon les actions sont exécutées lors des transitions entre les différents états ; ces états correspondent aux différentes situations se produisant lors de la vie du classificateur.

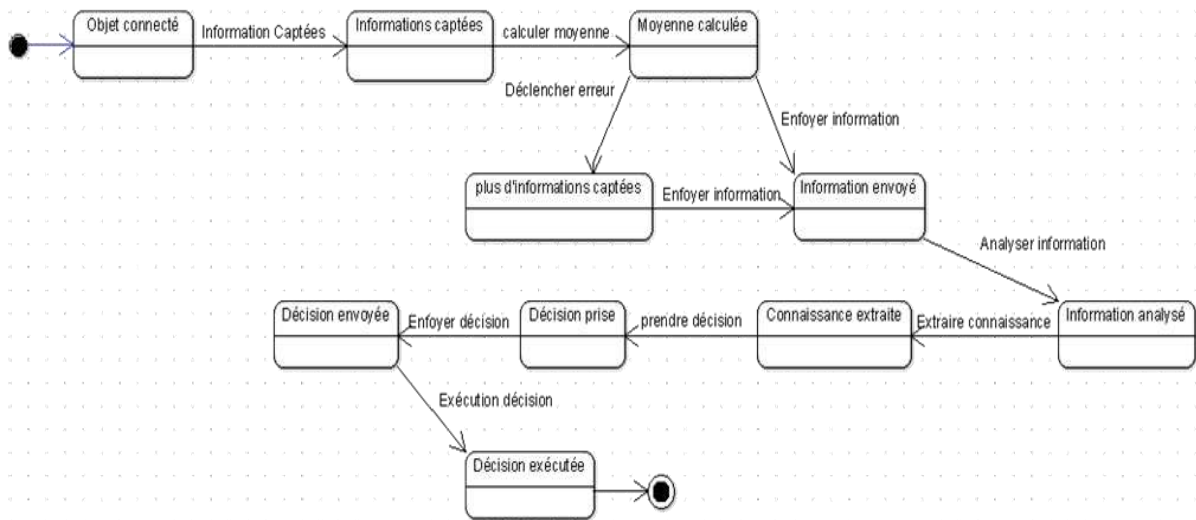


Figure 21 diagrammes d'états-transitions

Le diagramme d'états-transitions (figure21) montre les états globaux du système débutant par l'état initial l'objet connecté et finissant par l'état décision exécutée.

Conclusion

Dans ce chapitre, après nous avons parlé sur la conception générale de notre système. On a vu une proposition générale du problème qui ressemble à celui du jeu blackjack, pour la résolution on utilisera les algorithmes d'apprentissage par renforcement pour optimiser les énergies des objets connectés dans notre smarte homme.

Chapitre IV : Teste et réalisation

Introduction

Dans ce dernier chapitre, nous entamons la réalisation de notre application. Après l'étape de la conception définie au préalable, nous aborderons cette phase par la présentation et la définition des outils liés à la finalisation de l'application, enfin nous donnerons un aperçu des différentes testes réalisées.

1 Description de l'environnement

1.1 Langage de programmation utilisé

Pour développer notre application on a choisi le langage de programmation python

1.1.1 Python



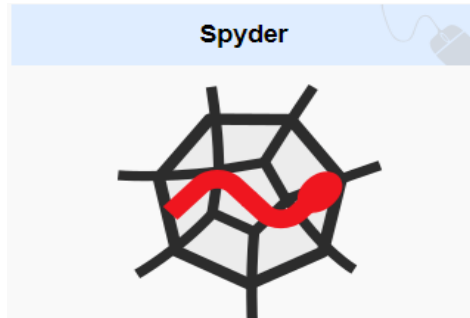
Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions .

Le langage Python est placé sous une licence libre proche de la licence BSD et fonctionne sur la plupart des plates-formes informatiques, des Smartphones aux ordinateurs centraux, de Windows à Unix avec notamment GNU/Linux en passant par MacOS, ou encore Androïde, Ios, et peut aussi être traduit en Java ou .NET. Il est conçu pour optimiser la

productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser.

1.2 Outils de développement

1.2.1 Spyder



Spyder est un environnement de développement pour Python, En comparaison avec d'autres IDE pour le développement scientifique, Spyder a un ensemble unique de fonctionnalités - multiplateforme, open-source, écrit en Python et disponible sous une licence non-copyleft. Spyder est extensible avec des plugins, comprend le support d'outils interactifs pour l'inspection des données et incorpore des instruments d'assurance de la qualité et d'introspection spécifiques au code Python.

2 Plateforme de test (la solution domotique)

La domotique est l'ensemble des technologies de l'électronique, de l'informatique et des télécommunications utilisées dans les habitations. La domotique vise à assurer des fonctions de sécurité (comme les alarmes), de confort (comme les volets roulants), de gestion d'énergie (comme la programmation du chauffage) et de communication (comme les commandes à distance) que l'on peut retrouver dans la maison. Il s'agit donc d'automatiser des tâches en les programmant ou les coordonnant entre elles.

3 Présentation de l'application

3.1 OpenAI gym environnement



Est une boîte à outils pour développer des algorithmes d'apprentissage par renforcement. Gym fournit une collection de problèmes de test appelés environnements pouvant être utilisés pour former un agent à l'aide d'un apprentissage par renforcement. Chaque environnement définit le problème d'apprentissage du renforcement que l'agent tentera de résoudre.

3.2 MC prediction_Sol

Au niveau de cette classe, on utilise l'algorithme de prédiction de Monte Carlo. Calcule la fonction de valeur pour une politique donnée en utilisant l'échantillonnage comme suivant :

Arguments:

- ✚ stratégie: une fonction qui associe une observation à des probabilités d'action.
- ✚ env: environnement OpenAI gym.
- ✚ num_episodes: Nombre d'épisodes à échantillonner.
- ✚ discount_factor: facteur de réduction gamma.

Résultats:

- ✚ Un dictionnaire qui mappe de l'état -> valeur.
- ✚ L'état est un tuple et la valeur est un float.

4 Test et validation

La validation du travail s'est faite sur un environnement domotique l'exécution doit être à partir du classe main qui la `classe smarthome`, on va tester trois catégories d'appareils qui

Chapitre IV : Teste et réalisation

sont : lampe (appareils à faible consommation énergétique), tv (appareil à moyenne consommation énergétique), réfrigérateur (appareil à fort consommation énergétique).avec l'algorithme de prédiction de Monte Carlo, les deux graphes (Figure 22)et (Figure 23) montre les courbe d'optimisation d'énergie consommé.

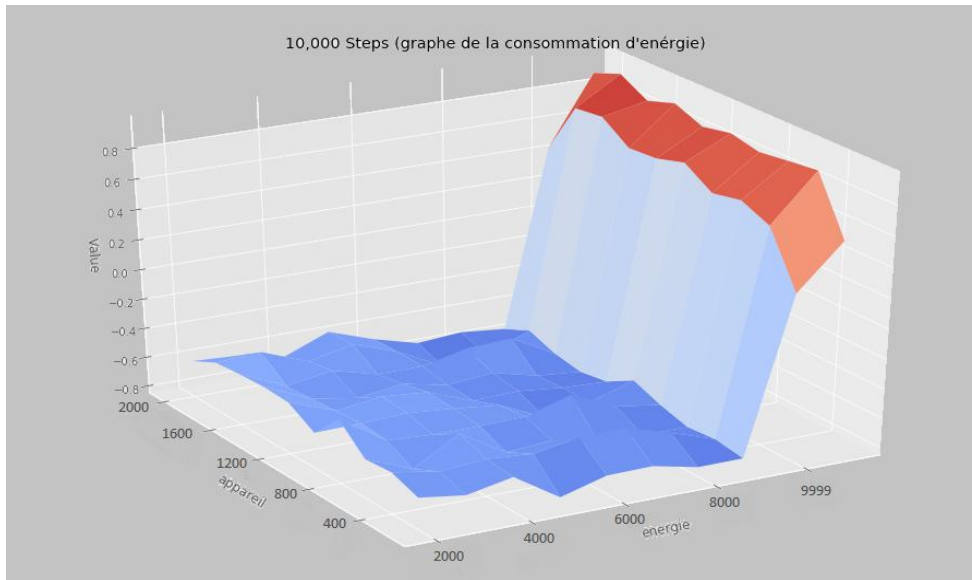


Figure 22 graphe de la consommation d'énergie avec 10000 épisodes

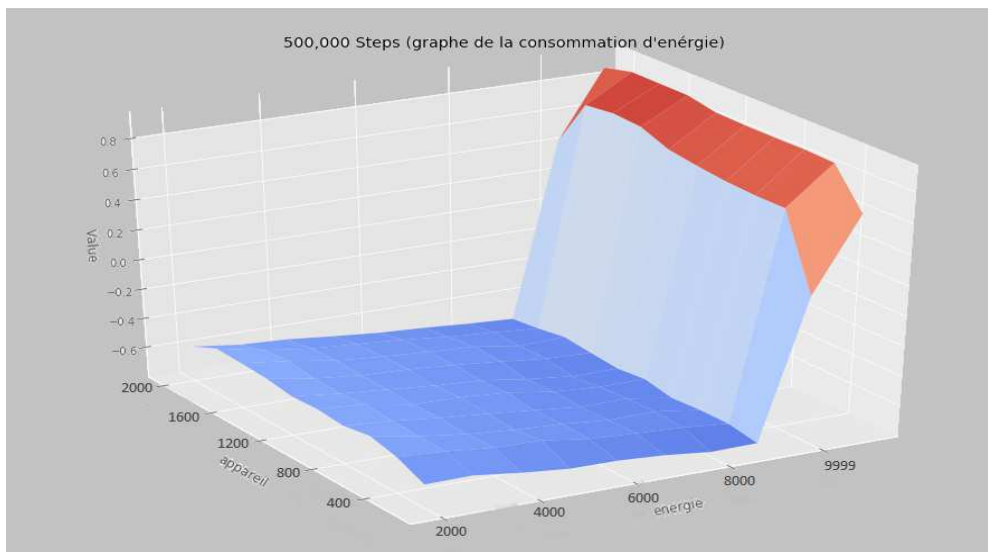


Figure 23 graphe de la consommation d'énergie avec 500000 épisodes

Chapitre IV : Teste et réalisation

Le première graphe montre la consommation énergétique optimisé qui ne dépasse pas le seuil qui égal à 10000watts, avec 10000 épisodes testées, l'axe AX représente la consommation du chaque appareil qui n'a doit être pas dépassé 2000watts, pour l'axe AY c'est la consommation énergétique total pour tout les appareils .Le deuxième graphe montre la même chose mais ici avec un grandes nombre d'épisodes. Donc on peut déduire de ce deux graphes qu'il ya une optimisation d'énergie.

Conclusion

Dans ce chapitre, nous avons présenté notre application développée et les résultats de test obtenus. Au début, nous avons présenté l'environnement de développement, le langage de programmation utilisé. Après nous avons parlé sur la plateforme de taste (solution domotique). Ensuite nous avons présenté notre application, gym environnement, mC prédiction. Finalement terminer avec une teste de validation qui est notre objective.

Conclusion générale

CONCLUSION GENERALE

Ce mémoire est le résultat de notre projet de fin d'études, Il consiste à concevoir et à implémenter une application logicielle graphique et interactive pour la résolution du problème de consommation d'énergie pour l'internet des objets. Pour remédier à cette travail, nous nous intéressons à faire une étude comparative des différentes approches existantes, puis proposer une nouvelle solution d'optimisation de consommation énergétique en exploitant l'apprentissage profond par renforcement avec la méthode de Monte Carlo.

Lors de la réalisation de ce projet, nous avons évoqué une classification des travaux proposés ces dernières années sur l'économie d'énergie, après nous avons essayé de donner une description générale pour le problème d'apprentissage par renforcement pour la résolution du problème consommation d'énergie qui ressemble à celui du blackjack. Le principal résultat obtenu, à partir d'exécution du code est sous forme des graphes (plotting).

Références bibliographiques

References bibliographies

- [Acevedo, P., Jiménez, M., & Cruz, A. \(2010\)](#). Simulation statique: méthode d'estimation de la puissance et de l'énergie dans des microprocesseurs intégrés, 53ème symposium international IEEE du Midwest sur les circuits et les systèmes ,pages 41–44.
- [Ahmed N. \(2014\)](#). L'intelligence ambiante et les systèmes de transport intelligents, Diplôme MAGISTER, Université Badji Mokhtar Annaba, Algérie.
- [Akyildiz, D., Pompili, S., & Melodia, T. \(2005\)](#). Défis de la recherche sur les réseaux de capteurs acoustiques: Réseaux ad hoc (Elsevier), 3(3), 257-279.
- [Akyildiz, T., Melodia, K., & Chowdhury, R. \(March 2007\)](#). Une enquête sur les réseaux de capteurs multimédias sans fil: Réseaux informatiques, Journal international des réseaux informatiques ET de télécommunications (Elsevier), 51(4), 921-960.
- [Acevedo, P., Jiménez, M., & Cruz, A., \(Aug 2010\)](#). Static simulation: Méthode d'estimation de la puissance et de l'énergie dans des microprocesseurs intégrés. International Midwest Symposium on Circuits and Systems, pages 41–44.
- [Benjamin B. \(2015\)](#). “Système de gestion de flux pour l'Internet des objets intelligents”. Thèse de doctorat, Université de Versailles Saint-Quentin-En-Yvelines.
- [Benoît P. \(2012\)](#). “Le Machine to machine : premier pas vers l'internet des objets”, rapport technique.
- [Benjamin B. \(2015\)](#). “Système de gestion de flux pour l'Internet des objets intelligents”. Thèse de doctorat, Université de Versailles Saint-Quentin-En-Yvelines.
- [Bazzaz, M., Salehi, A., & Ejlali A. \(July 2013\)](#). Un modèle d'estimation d'énergie précis au niveau de l'instruction et un outil pour les systèmes embarqués. Transactions sur instrumentation et mesure, 62(7):1927–1934.
- [Charline, D. \(2011\)](#). Les produits blancs connectés ont-ils un avenir auprès des particuliers ?”, Thèse professionnelle, MBA spécialisés et formation continue du pôle universitaire léonard de Vinci.
- [Charline D. \(2011\)](#). Les produits blancs connectés ont-ils un avenir auprès des particuliers ?, *Thèse* professionnelle, MBA spécialisés et formation continue du pôle universitaire léonard de Vinci.

Références bibliographiques

- [Charith, P., & al. \(2014\)](#). Prise en compte du contexte informatique pour les objets connectés: Journal des enquêtes et tutoriels sur les communications, 16(1), pp. 414 – 454.
- [Carroll, A., & Heiser G. \(2010\)](#). Une analyse de la consommation d'énergie dans un téléphone intelligent, Conférence technique, pages 21–21, Berkeley.
- [Camille B. 2005](#), D.A.M.A.S. U. Laval.
- [Clément C. \(novembre 2003\)](#). Les multi layer perceptron (MLP).
- [Daniele, M., Sabrina, S., Francesco, D., & Imrich Ch. \(2012\)](#). Internet des objets: applications de la vision et défis de la recherche”, Journal des réseaux ad hoc, 10(7), pp.1497-1516.
- [Dutech A. \(1999\)](#). Apprentissage d'environnement : approches cognitives et Comportementales, Thèse de doctorat, Ecole Nationale supérieure de l'aéronautique et de L'espace, Toulouse.
- [David F. \(2004\)](#). Robotique Mobile.
- [Farida R. \(2019\)](#).Gestion de contexte pour l'internet des objets, These de doctorat, Université Mohamed khider,Biskra
- [Ghofrane F. \(2015\)](#). “Middleware pour les IoT”. Conférence internationale sur l'informatique distribuée dans les systèmes de capteurs (DCOSS), pp. 100–112, Tunisie.
- [Gérald, S., & Sébastien, L. \(2008\)](#). L'Internet des objets en 2020: une feuille de route pour l'avenir, Rapport d'atelier.
- [Glo P. \(2015\)](#). Algorithmes d'apprentissage pour les systèmes d'inférence floue.
- [Hwang, S., Ahn, Y., & Chung K. \(Sept 2010\)](#). Technique de prévision de la consommation d'énergie pour un appareil mobile intégré utilisant un schéma de décharge de la batterie. Conférence sur l'infrastructure de réseau et le contenu numérique, pages 907–910.
- [Homb R. \(2016.\)](#). Adaptive Store ET Forward.
- [Jean-Paul, A., & al. \(2014\)](#). “Projet INCOME: Infrastructure de gestion de COntexte Multi-Echelle pour l'Internet des Objets”, In Conférence Francophone sur les Architectures Logicielles (CAL), pp. 1–2, Université de Toulouse.

Références bibliographiques

- [Jing, Q., Athanasios, V., Jiafu, W., Jingwei, Lu., & Dechao Qiu. \(2014\).](#) “Sécurité de l'Internet des objets: perspectives ET défis”, *Journal des réseaux sans fil*, 20(8), pp. 2481–2501.
- [Jalali, K., Hinton, R., Ayre, T., Alpcan, R., & Tucker S. \(May 2016\).](#) Le brouillard informatique peut aider à économiser de l'énergie dans le Cloud computing. *Journal de l'IEEE sur certains domaines des communications*, 34(5):1728–1739.
- [Jun K. \(2015\).](#) Architecture de gestion sécurisée et efficace pour l'IoT, *Ecole d'Informatique et d'Ingénierie*, pp. 499-500, UNSW Australie.
- [Kechar B. \(2010\).](#) Problématique de la consommation d'énergie dans les réseaux De capteurs sans fil, *These de doctorat, Université d'oran,Oran*.
- [Laurent, J., Nathalie, J., Eric, S., & Eric M. \(2004\).](#) Analyse de la puissance au niveau fonctionnel: une approche efficace pour modéliser la consommation de puissance de processeurs complexes. Dans les actes de la conférence sur le design, pages 10666.
- [Mateu, F., Moll, N. \(2005\).](#) Examen des techniques et applications de récupération d'énergie pour la microélectronique. *la Société internationale d'ingénierie optique*, Vol (5837), pp. 359-373.
- [Mégarien, M., & Potkonjak V. \(2003\).](#) Réseaux de capteurs sans fil, pp. 2990-2996.
- [Malek N. \(2014\).](#) “L'intelligence ambiante et les systèmes de transport intelligents”, *Diplôme MAGISTER*, Université Badji Mokhtar Annaba, Algérie.
- [Milenkovic, M., Milenkovic, E., Jovanov, D., Hite, P., & Raskovic D. \(March 2005\).](#) Environnement de surveillance de l'alimentation au moment de l'exécution des plates-formes de réseaux de capteurs sans fil, pages 406–410.
- [Mijovic, R., Cavallari, Z &, Buratti C. \(Déc. 2015\).](#) Caractérisation expérimentale de la consommation d'énergie dans les réseaux corporels: dans l'Internet des objets (*WF-IoT*), pages 514–519.
- [Munos R. \(2007\).](#) Transparents cours "Apprentissage par Renforcement", *Master Mathématiques, Vision, Apprentissage, ENS Cachan*.
- [Mezaache H. \(2008\).](#) Les réseaux de Neurons formels Et Les systèmes Neuro-Flous Pour l'apprentissage par renforcement, *Diplôme MAGISTER*, Université El Hadj Lakhdar Batna, Algérie.
- [Ovidiu, V., & al. \(2011\).](#) la recherche stratégique sur l'Internet des objets. Dans V.Ovidiu & F.Peter (édit).*Internet des objets - Technologies convergentes pour environnements intelligents et écosystèmes intégrés*, pp.10-50.

Références bibliographiques

- [Pierrick M. \(2015\)](#). “Gestion de bout en bout de la Qualité de Contexte pour l’Internet des Objets: le cadriciel QoCIM”, Thèse doctoral, Université Toulouse.
- [Pascal G. \(2004\)](#). Exploration guidée et induction de comportement géométrique en Apprentissage par renforcement, Thèse doctorat juillet.
- [Raja B. \(2015\)](#). “Déploiement de systèmes répartis multi-échelles processus, Thèse Doctoral, Université de Toulouse.
- [Rice, A., & Hay S. \(Décembre 2010\)](#). Mesure de la consommation d'énergie des téléphones mobiles pour les réseaux sans fil 802.11, 6(6):593–606.
- [Sakamura L. \(November 2005\)](#). T-Engine la plate-forme ouverte de l'ère de l'informatique omniprésente: revue internationale de Circuits, pp. 3-6.
- [Shnayder, V., Mark, H., Bor-rong, C., Geoff W, W& Matt W. \(2004\)](#). Simulation de la consommation d'énergie d'applications de réseau de capteurs à grande échelle, pages 188–200, New York.
- [Vercouter, L.,Beaune,P., & Sayettat N. \(1998\)](#). Apprentissage dans les sma. In actes des journées française sur l’intelligence artificielle distribuée et les systèmes multi-agents (JFIADSMA’98) Hermès, Paris France.
- [Willig A. \(2006\)](#). Réseaux de capteurs sans fil: défis de concept et approches. Dans S. Verlag, pp.224- 231.
- [Xin, L., Martina, E., José, F., Gregorio R. \(2015\)](#). “Architectures de middleware conscientes du contexte: Enquête et défis”: Journal des capteurs, 15(8), pp. 20570-20607.
- [Yu, V., Prasanna, K ., & Krishnamachari, B.\(2006\)](#). Traitement de l'information et routage dans les réseaux de capteurs sans fil: Édition scientifique mondiale, pp.1-21.
- [Yick, B., Mukherjee, D., & Ghosal, X. \(2008\)](#). Enquête sur le réseau de capteurs sans fil. Réseaux ad hoc (Elsevier), 52(12), 2292-2330.
- [Yen-Kuang Ch. \(2012\)](#). Défis et opportunités de l'Internet des objets, pp.383-388, Australia.
- [Yeh C. \(May 2008\)](#) .Techniques de reconfiguration .Dynamics pour les réseaux de capteurs sans fil, Mémoires fin d’études, Université de Massachusetts, USA.
- [Youssef Z. \(2004\)](#). Apprentissage par renforcement et systèmes distribués, Thèse doctorat juillet.