



HAL
open science

Une approche centrée sur les contenus pour les réseaux de capteurs

Ghada Jaber

► **To cite this version:**

Ghada Jaber. Une approche centrée sur les contenus pour les réseaux de capteurs. Networking and Internet Architecture [cs.NI]. Université Paul Sabatier - Toulouse III, 2018. English. NNT : 2018TOU30196 . tel-04505054

HAL Id: tel-04505054

<https://theses.hal.science/tel-04505054>

Submitted on 14 Mar 2024

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

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



THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Paul Sabatier - Toulouse 3*

Présentée et soutenue le *23/11/2018* par :

GHADA JABER

A Content-Centric Approach for Wireless Sensor Networks

JURY

| | | |
|-----------------------|--|-------------|
| OLIVIER FESTOR | Professeur à Télécom Nancy | Rapporteur |
| YACINE GHAMRI-DOUDANE | Professeur à Université de la Rochelle | Rapporteur |
| HOUDA LABIOD | Professeur à Télécom ParisTech | Examinateur |
| TOUFIK AHMED | Professeur à INP Bordeaux | Examinateur |
| ZOUBIR MAMMERI | Professeur à Université Paul Sabatier | Examinateur |
| THIERRY GAYRAUD | Professeur à Université Paul Sabatier | Examinateur |
| RAHIM KACIMI | Maître de Conférences à Université Paul Sabatier | Examinateur |

École doctorale et spécialité :

MITT : Domaine STIC : Réseaux, Télécoms, Systèmes et Architecture

Unité de Recherche :

Institut de Recherche en Informatique de Toulouse (IRIT)

Directeur(s) de Thèse :

Thierry GAYRAUD et Rahim KACIMI

Rapporteurs :

Olivier FESTOR et Yacine GHAMRI-DOUDANE

Acknowledgment

I would like to thank Mr. Thierry Gayraud for the precious time, advice and opportunities that he offered to me during these three years of thesis. I am also grateful to Mr. Rahim Kacimi for offering me the master internship and then for proposing to me this thesis opportunity. Mr. Kacimi knew how to trust me and leave me all the freedom to carry out my work while accompanying me with his advice. I thank him very sincerely for his personal and professional investment to make me move forward and overcome numerous obstacles I have been facing through my research.

I thank Mr Yassime Gamri-Doudane and Mr Olivier Festor for accepting to review my manuscript and for giving me valuable advice. I want also to thank all the jury members for their their insightful comments and encouragement.

I am also grateful to my neOCampus family especially to Ms Marie-Pierre Gleizes for her support and for providing me an opportunity to join this family.

I would like to thank my colleagues Malik and Usman for their support and for their good mood during the difficult moment. I also thank all the IRIT laboratory staff that I have been able to work with. I want to thank Mahdi, Amani, Adel, Amine, Mohamed, Mahmoud, Frank, Paul, Zeineb, Hung, and Thiziri for their friendship, for their support, and for the coffee break and lunches that we shared together during these three years.

I would like to thank my Italian team who hosted me for my mobility internship including Mr. Alfredo Grieco, Agnese, Pierro, Pierro, Mauro, and Daniele. I thank also Linda, Nousa, Mouldi, John, and Skander who shared this experience with me.

Manel, Eya, Amani, Anouar, Azzouz, Mariem, Ameni, Mouna, Amine, Walid, Sameh, Balkis, Abdelaziz, Mariem, Hamza, Alexandra, Chiraz, Ines, Bilel, Mehdi, Hamdi, Said, Dhouha, Sonia, and Habib, I could never thank you enough for supporting me so much and believing in me. You were here in the good and the bad moments, you were my pillars. You are my second family! I love you and I can't imagine this adventure without you. Thank you for being here for me and thank you for illuminating my life.

I am also grateful to my RC2T and GRAVETTE family who supported me during my thesis and believed in me.

My ultimate thanks are to my family especially to my mother Kmar who never stopped encouraging and supporting me. Thank you for sacrificing your life for me, I will never be able to give you back what you did for me. I would like also to thank my aunts Moufida, Zeineb, Kalthoum, and Kaouther for their help, love and support. I thank my uncle Mounir for always being there for me. I thank also my cousin Hela, Haifa, and Hiba for being the sisters that I have ever had. And finally, I want to thank my grandfather Jomaa and my grandmother Halima.

'Without the precious support of all of you, it would not be possible to conduct this research. I would never thank you enough.'

Abstract

Wireless sensor networks are an essential part of the Internet of Things (IoT) "perception" layer. IoT connects the digital world created by conventional computer networks to the physical world. They continually bring new applications to life through a large number of elements that collect, process and disseminate environmental data.

Today, moves accross the IoT a large and varied volume of data. Data is generated in a continuous way with a greater emphasis on information and not on its source. This indifference to the source is reinforced by the interchangeable and redundant deployments of the sensor networks.

In this thesis, we focus on integrating the principles and mechanisms of content-oriented networks in wireless sensor networks to improve the operation and performance of these networks. Hence, we first focused on the temporal relevance of data in content-centric sensor networks. Indeed, we considered the content lifetime (or freshness) and we proposed two approaches (one reactive and the other proactive) for their update.

In the second part of the thesis, we proposed a mechanism based on the control of the duty-cycle to overcome the impact of the flooding mainly used to disseminate the interests sent by the users and the corresponding contents. For this purpose, we tried to maintain a sufficient subset of nodes necessary to satisfy the interests received by the network. The main challenge was to reduce energy consumption thanks to a mechanism controlling the node activity while keeping a good interest satisfaction rate.

Finally, to improve the content caching in a sensor network, we have studied the existing strategies and identified the parameters impacting their performance. We then proposed a strategy placing the contents according to the degree of the nodes and their distance from the source. An exhaustive comparative study

with other solutions show that the proposed mechanisms guarantee good performance in terms of latency, energy consumption and interest satisfaction rate.

Keywords— Internet of Things, wireless sensor networks, information-centric networking, content-centric networking, data freshness, flooding, forwarding, duty-cycling, in-network caching.

Resumé

Les réseaux de capteurs sans fil constituent une partie essentielle de la couche " perception " de l'Internet des objets (IoT ou *Internet of Things*), et reconnectent le monde numérique créée par les réseaux informatiques classiques au monde physique. Ils font émerger sans cesse de nouvelles applications grâce à un nombre important d'éléments, qui collectent des données de l'environnement, les traitent et les disséminent.

Aujourd'hui, circulent dans l'IoT des volumes de données importants, très variés et souvent générés de façon continue mettant plus l'accent sur l'information et non sur sa source. Cette indifférence sur la source est renforcée par les déploiements interchangeable et redondants des réseaux de capteurs. Dans cette thèse, nous nous attachons à l'intégration des principes et mécanismes des réseaux orientés contenus dans les réseaux de capteurs sans fil afin d'améliorer le fonctionnement et les performances de ces derniers. Nous nous sommes intéressés, dans un premier temps, à la pertinence temporelle des données dans les réseaux de capteurs centrés sur les contenus. En effet, nous avons opté pour la prise en considération de la durée de vie (ou fraîcheur) des contenus et proposé deux approches (une réactive et l'autre proactive) pour leur mise à jour.

Dans la seconde partie de la thèse, nous avons proposé un mécanisme fondé sur le contrôle du duty-cycle afin d'atténuer l'impact de l'inondation principalement utilisée pour disséminer les intérêts envoyés par les utilisateurs et les contenus correspondant. Pour cela, nous avons cherché à maintenir un sous-ensemble suffisant de nœuds nécessaires à la satisfaction des intérêts reçus par le réseau. Le défi principal était de réduire la consommation d'énergie grâce à un mécanisme d'endormissement/réveil des nœuds capteurs tout en gardant un bon taux de satisfaction des requêtes.

Enfin, pour améliorer la mise en cache des contenus dans un réseau de capteurs, nous avons étudié les stratégies existantes et recensé les paramètres impactant leur performance. Nous avons ensuite proposé une stratégie plaçant les contenus en fonction du degré des nœuds et leur distance de la source. Une campagne de simulations et des comparaisons avec d'autres solutions, montrent que les mécanismes proposés garantissent de bonnes performances en termes de latence, de consommation d'énergie et de taux de satisfaction des intérêts.

Keywords— Internet des Objets, réseaux de capteurs, réseaux centrés sur les informations, réseaux centrés sur les contenus , fraîcheur des données, diffusion, *duty-cycling*, *caching*.

Glossary

| | |
|------------|---|
| 6LowPAN | IPv6 Low power Wireless Personal Area Networks |
| ADDC-CCWSN | Adaptive and fully Distributed Duty-Cycle for Content-Centric Wireless Sensor Network |
| AHP | Analytic Hierarchy Process |
| CCN | Content Centric Networking |
| CDN | Content Delivery Networks |
| CO | Content Object |
| CoAP | Constrained Application Protocol |
| CPU | Central Processing Unit |
| CS | Content Store |
| CSDD | Caching Strategy Distance and node Degree aware |
| CSMA | Carrier Sense Multiple Access |
| DFCCN-WSN | Data Freshness aware Content-Centric Networking in Wireless Sensor Networks |
| DMIF | Dual Mode Interest Forwarding scheme |
| DONA | Data-Oriented Network Architecture |
| FIB | Forwarding Information Table |
| GAF | Geographic Adaptive Fidelity |
| GCCS | Global Cluster Cooperation for wireless Sensor networks |
| GEAR | Geographical and Energy Aware Routing |

| | |
|-------------|---|
| ICN | Information Centric Networking |
| IEEE | Institute of Electrical and Electronics Engineers |
| IETF | Internet Engineering Task Force |
| IoT | Internet of Things |
| IPv6 | Internet Protocol version 6 |
| MAC | Medium Access Control |
| MEMS | MicroElectroMechanical Systems |
| NetInf | Network of Information |
| NDN | Named Data Networking |
| NSF | National Science Foundation |
| PARC | Palo Alto Research Center |
| PDLP | Packet Diffusion-Limited Protocol |
| PDFCCN-WSNs | Proactive Data Freshness aware Content-Centric Networking in Wireless Sensor Networks |
| PIT | Pending Interest Table |
| PSIRP | Publish-Subscribe Internet Technology |
| QoS | Quality of Service |
| RAM | Random Access Memory |
| RDFCCN-WSNs | Reactive Data Freshness aware Content-Centric Networking in Wireless Sensor Networks |
| RFID | Radio Frequency IDentification |
| ROM | Read-Only Memory |
| RPL | Routing Protocol for Low-Power and Lossy Networks |
| SAR | Secure-aware Ad hoc Routing Protocol |
| SPIN | Sensor Protocol for Information via Negotiation |
| TCP | Transmission Control Protocol |
| TEEN | Threshold sensitive Energy Efficient sensor Network protocol |
| TLS | Transport Layer Security |

| | |
|------|------------------------------------|
| UDP | User Datagram Protocol |
| WMSN | Wireless Multimedia Sensor Network |
| WSAN | Wireless Sensor and Actor Network |
| WSN | Wireless sensor network |
| XIA | eXpressive Internet Architecture |

Table des matières

| | |
|--|----------|
| Acknowledgment | iii |
| Abstract | v |
| Resumé | vii |
| Glossary | ix |
| 1 Introduction | 1 |
| 1.1 Context | 2 |
| 1.2 Motivations | 2 |
| 1.3 Problem statement | 4 |
| 1.4 Contributions | 5 |
| 1.5 Manuscript outline | 5 |
| 2 State of Art | 7 |
| 2.1 Introduction | 8 |
| 2.2 Internet of Things | 8 |
| 2.2.1 Architecture of the Internet of Things | 9 |
| 2.3 Wireless Sensor Networks | 10 |
| 2.3.1 Architecture of a sensor node | 10 |
| 2.3.1.1 Sensing unit | 11 |

| | | |
|---------|---|----|
| 2.3.1.2 | Processing unit | 11 |
| 2.3.1.3 | Transceiver unit | 11 |
| 2.3.1.4 | Power unit | 12 |
| 2.3.2 | Energy conservation techniques in WSNs | 12 |
| 2.3.2.1 | Duty-cycling techniques | 12 |
| 2.3.2.2 | Data driven techniques | 13 |
| 2.3.2.3 | Mobility techniques | 13 |
| 2.3.3 | Routing protocols in wireless sensor networks | 14 |
| 2.3.3.1 | Taxonomy of routing protocols | 14 |
| 2.3.4 | WSN as a part of the IoT | 15 |
| 2.3.5 | Challenges of integration of WSNs in IoT | 16 |
| 2.4 | Information-Centric Networking Paradigm | 17 |
| 2.4.1 | The evolution of the Internet and the motivation behind ICN | 17 |
| 2.4.2 | Difference between host-centric and data-centric | 18 |
| 2.4.3 | ICN features | 18 |
| 2.4.3.1 | Naming | 18 |
| 2.4.3.2 | Routing | 18 |
| 2.4.3.3 | Caching | 19 |
| 2.4.3.4 | Mobility | 19 |
| 2.4.3.5 | Security | 19 |
| 2.4.4 | ICN limitations | 20 |
| 2.4.4.1 | Caching | 20 |
| 2.4.4.2 | Routing and Forwarding | 20 |
| 2.4.4.3 | Congestion control | 20 |
| 2.4.4.4 | Security | 21 |
| 2.5 | Which ICN paradigm for WSNs? | 21 |
| 2.5.1 | Publish and subscribe | 21 |
| 2.5.2 | NDN | 22 |

| | | |
|----------|--|-----------|
| 2.5.3 | CCN | 23 |
| 2.6 | Conclusion | 24 |
| 3 | Content-Centric Approach for Wireless Sensor Networks: Suitability? | 27 |
| 3.1 | Introduction | 28 |
| 3.2 | Content-centric networking approach | 28 |
| 3.3 | Content-centric networking in wireless sensor networks | 30 |
| 3.3.1 | Related work | 30 |
| 3.3.2 | Node structure | 31 |
| 3.3.3 | CCN entities | 33 |
| 3.3.4 | CCN stack in WSNs | 33 |
| 3.3.5 | Advantages of CCN integration in WSNs | 34 |
| 3.3.5.1 | Content caching | 34 |
| 3.3.5.2 | Naming strategy | 35 |
| 3.3.5.3 | Scalability | 35 |
| 3.3.5.4 | Mobility | 35 |
| 3.3.5.5 | Forwarding | 35 |
| 3.3.5.6 | Security | 36 |
| 3.3.6 | Content-Centric integration in Wireless Sensor network | 36 |
| 3.4 | Cache freshness | 37 |
| 3.5 | Routing and Forwarding | 38 |
| 3.6 | In-network caching | 40 |
| 3.7 | Application scenario: traffic on Paul Sabatier University | 42 |
| 3.8 | Conclusion | 44 |
| 4 | Data Freshness in Content-Centric Wireless Sensor Networks | 45 |
| 4.1 | Introduction | 46 |
| 4.2 | State of the art | 46 |
| 4.2.1 | Data freshness and reliable data delivery in WSNs | 46 |

| | | |
|----------|---|-----------|
| 4.2.2 | Reactive and proactive routing in Wireless Sensor Networks | 48 |
| 4.3 | Content lifetime in CCN for WSNs | 49 |
| 4.3.1 | Principle of DFCCN-WSNs | 50 |
| 4.3.2 | Motivating example | 50 |
| 4.4 | Content-update strategies | 52 |
| 4.4.1 | Proactive approach | 53 |
| 4.4.2 | Reactive approach | 54 |
| 4.5 | Performance evaluation | 57 |
| 4.5.1 | Simulation set-up | 57 |
| 4.5.2 | Evaluation metrics | 57 |
| 4.5.3 | Simulation results | 58 |
| 4.5.3.1 | One user | 58 |
| 4.5.3.2 | Multiple users | 59 |
| 4.6 | Conclusion | 63 |
| 5 | A Duty-cycling Approach for Content-Centric Wireless Sensor Networks | 65 |
| 5.1 | Introduction | 66 |
| 5.2 | Duty-cycling schemes for WSNs | 66 |
| 5.2.1 | Definition of the duty-cycle | 66 |
| 5.2.2 | Taxonomy of duty-cycling schemes | 67 |
| 5.2.2.1 | Synchronous schemes | 67 |
| 5.2.2.2 | Semi-synchronous schemes | 68 |
| 5.2.2.3 | Asynchronous schemes | 68 |
| 5.3 | Energy models in CCN | 69 |
| 5.4 | System Model | 70 |
| 5.4.1 | Proposed CCN node model | 70 |
| 5.4.2 | Proposed energy model | 73 |
| 5.4.3 | Numerical analysis of the energy model | 77 |

| | | |
|----------|---|-----------|
| 5.5 | Proposed duty-cycle plan | 78 |
| 5.5.1 | Principle | 78 |
| 5.6 | Performance Analysis | 80 |
| 5.6.1 | Simulation set-up | 80 |
| 5.6.2 | Evaluation metrics | 82 |
| 5.6.3 | Simulation results | 82 |
| 5.6.3.1 | Study of the threshold impact | 82 |
| 5.6.4 | Discussion | 87 |
| 5.7 | Conclusion | 88 |
| 6 | Content Popularity Aware Caching Strategies in WSNs | 91 |
| 6.1 | Introduction | 92 |
| 6.2 | State of the art | 92 |
| 6.2.1 | Data caching in WSNs | 92 |
| 6.2.2 | Content caching in CCN | 93 |
| 6.2.2.1 | Interest routing in CCN | 94 |
| 6.2.2.2 | Cache placement strategies | 94 |
| 6.2.2.3 | Cache replacement policies | 96 |
| 6.3 | A collaborative caching strategy for content-centric enabled wireless sensor networks | 98 |
| 6.3.1 | Content request process | 98 |
| 6.3.2 | Cache admission control | 99 |
| 6.3.2.1 | Proposed cache placement approach | 100 |
| 6.3.3 | Cache replacement policy | 102 |
| 6.4 | Performance evaluation | 104 |
| 6.4.1 | Simulation set-up | 104 |
| 6.4.2 | Evaluation metrics | 105 |
| 6.4.3 | Simulation results | 106 |
| 6.4.4 | Discussion | 116 |

| | |
|---------------------------------------|------------|
| 6.5 Conclusion | 117 |
| 7 Conclusions and Perspectives | 119 |
| 7.1 Conclusion | 119 |
| 7.2 Contributions summary | 120 |
| 7.3 Perspectives | 121 |
| List of publications | 123 |
| Bibliography | 141 |

Table des figures

| | | |
|-----|--|----|
| 2.1 | An example of an IoT protocol architecture inspired from [1]. | 10 |
| 2.2 | Components of a sensor node inspired from [2]. | 11 |
| 2.3 | Classification of energy conservation schemes. | 12 |
| 2.4 | Publish and subscribe paradigm [3]. | 22 |
| 2.5 | NDN/CCN Information routing. | 23 |
| 3.1 | IP/CCN stack inspired from [4]. | 29 |
| 3.2 | CCN packets inspired from [4]. | 29 |
| 3.3 | CCN node structure elaborated for a sensor node and inspired from [5]. | 32 |
| 3.4 | CCN operations, interest dissemination and content object delivery. | 33 |
| 3.5 | An example of a CCN stack in WSNs. | 34 |
| 3.6 | CCN enabled WSN communication. | 36 |
| 3.7 | Wireless sensor network deployment and CCN communications in a campus. | 43 |
| 4.1 | Interests broadcast in CCN-WSN network. | 51 |
| 4.2 | Expressing interests for temperature and humidity. | 51 |
| 4.3 | The user recovers the content from intermediate nodes. | 52 |
| 4.4 | When the node $N8$ realizes that the data lifetime is expired, it broadcasts an interest. | 53 |
| 4.5 | When the node $N13$ receives an interest for the queue length, it responds with this content. | 54 |
| 4.6 | The user broadcasts an interest to have an idea about the queue state in front of the restaurant. | 55 |
| 4.7 | The content is recovered from the source node $N13$ | 55 |
| 4.8 | Delay for different users during the 1 st round for different types of generated content. | 60 |

| | | |
|------|---|-----|
| 4.9 | Delay for different users during the 2^{nd} round with the reactive approach RDFCCN-WSNs. | 61 |
| 4.10 | Delay for different users during the 2^{nd} Round with the proactive approach PDFCCN-WSNs. | 61 |
| 4.11 | Energy Consumed by the network for different approaches for 2 rounds. | 62 |
| 5.1 | An example of a duty-cycle of 20%. | 67 |
| 5.2 | Taxonomy of duty-cycling schemes. | 67 |
| 5.3 | A CCN node model with three tables CS, FIB, and PIT. | 72 |
| 5.4 | Energy consuming operations. | 73 |
| 5.5 | Energy consumption vs. p_i , λ_{DC} , and τ_i . | 78 |
| 5.6 | Energy consumption vs. network size for a threshold 20%. | 83 |
| 5.7 | Interest satisfaction rate for different network sizes with a threshold equal to 20% | 84 |
| 5.8 | Energy consumption for threshold 50% vs. network size. | 85 |
| 5.9 | Energy consumption for 80 nodes and the threshold 50%. | 85 |
| 5.10 | Interest satisfaction rate for threshold 50% and different network sizes. | 86 |
| 5.11 | Mean delay realized by different schemes for both thresholds. | 87 |
| 5.12 | Packet loss ratio for both thresholds 20 % and 50 %. | 87 |
| 6.1 | Caching placement policies. | 96 |
| 6.2 | Least Frequently Used replacement policy: once the cache is full LFU replaces the least frequently used (frequencies calculated by bits in the data) by most frequented data. | 97 |
| 6.3 | Least Recently Used replacement policy: once the cache is full LRU replaces the least recently used (identified by bits in the data) by most used data. | 97 |
| 6.4 | FIFO replacement policy: First In First Out replacement policy. | 98 |
| 6.5 | Zipf interest distribution depending on the value of α varied from 0.2 to 1.8 for 20 different generated contents. | 99 |
| 6.6 | The first condition of our caching scheme: distance from the source node. | 101 |
| 6.7 | An example of function of our strategy when Δ is equal to 50% and $x > 3$. | 102 |
| 6.8 | Percentage of nodes detaining a certain node degree. | 105 |
| 6.9 | Energy consumption for all the strategies when using FIFO and Popularity-based replacement policies for 80 nodes. | 107 |

| | |
|---|-----|
| 6.10 Stretch for all the strategies when using LCE and Popularity-based replacement policies. . . | 109 |
| 6.11 Network lifetime for different α for all the strategies when using a FIFO replacement policy. | 110 |
| 6.12 Network lifetime for different α for all the strategies when using a Popularity-based replacement policy. | 111 |
| 6.13 Cache hit for all the strategies when using FIFO and Popularity-based replacement policies. . | 112 |
| 6.14 Replacement rate for all the strategies when using FIFO and Popularity-based replacement policies. | 113 |
| 6.15 Diversity for different α for all the strategies when using a FIFO replacement policy. | 114 |
| 6.16 Diversity for different α for all the strategies when using a Popularity-based replacement policy. | 115 |

Liste des tableaux

| | | |
|-----|---|-----|
| 3.1 | Studied CCN features in several CCN enabled WSNs works. | 32 |
| 4.1 | Simulation parameters | 58 |
| 4.2 | Delay when applying the RDFCCN-WSNs update strategy | 58 |
| 4.3 | Delay when applying the PDFCCN-WSNs update strategy | 59 |
| 5.1 | List of mathematical symbols | 71 |
| 5.2 | Analysis parameters | 77 |
| 5.3 | Simulation parameters | 81 |
| 6.1 | Simulation parameters | 104 |

CHAPITRE 1

Introduction

Contents

| | | |
|-----|------------------------------|---|
| 1.1 | Context | 2 |
| 1.2 | Motivations | 2 |
| 1.3 | Problem statement | 4 |
| 1.4 | Contributions | 5 |
| 1.5 | Manuscript outline | 5 |

1.1 Context

Internet of Things (IoT) is a new paradigm contributing in changing the world that will interconnect more than 25 billion of heterogeneous constrained devices belonging to heterogeneous networks by 2020 according to IBM [6]. In the future, the IoT architecture including Wireless Sensor Networks (WSNs) promises to be widely deployed in real environments. Indeed, IoT detains a significant role in facilitating connectivity of WSNs enabling an interaction between the physical and virtual world. This results in creating an ability to allow flexibility within the physical system and in adapting with the service requirements. Therefore, WSN has been one of the main technology that brings IoT closer to reality by enabling autonomous and intelligent links. In addition to that, the environment where sensor networks are deployed to collect information is compliant with the IoT expectation. Furthermore, WSNs have found room for application in several domains such as environmental monitoring, smart cities, agriculture, and healthcare. In the last years, WSNs have witnessed a large interest from the research community who aims to enhance their performances and communication efficiency.

The small, inexpensive and low powered WSN sensors will bring to the IoT objects reasonable costs and an ability to be installed anywhere. On the other hand, their integration into IoT will constitute a major evolution of WSNs. Additionally, as the modern world shifts to this new age of WSNs in the IoT, the manner the data is collected will present an issue to which a solution has to be found. Indeed in IoT, the communication is based on the data exchanged between the devices. Then, challenges have also been concluded from the perspective of the data-centric and energy-efficient approaches.

1.2 Motivations

The IoT interconnected devices are increasing day per day generating a traffic proliferation. The spreading deployment may also result in the coexistence of multiple kinds of entities hard to manage. So, the coordination between these entities is a challenging issue. As already stated, thanks to the sensors low cost, wireless sensor networks present a good alternative to enable IoT applications because they detain a certain manner to manage data exchanged between devices. However, the model for communication in wireless sensor networks does not always follow the sender/receiver model of the Internet [7]. There are many applications for which traditional routing does not fit. For instance, let imagine a scenario in which a sensor

node wants to request an information from a location where an interesting event is occurring (e.g., an information about the number of students doing the queue in front of the restaurant). The requester does not necessarily want to request information from a particular node (e.g., by an ID or IP address), but rather from any node that can provide the data. Therefore, it was noticed that a mechanism whereby nodes can request, advertise and disseminate data to interested parties, is needed [8]. This motivated the development of data-centric approaches to networking in WSNs.

To face this new demand, the Information-Centric Networking (ICN) paradigm [9] emerged in the third millennium. ICN grounds networking primitives on content names rather than node locators (as in the current Internet). ICN targets seamless mobility, native multicast/multipath support, and content-oriented security to better reflect the needs of today users.

Moreover, ICN could greatly improve the efficiency of content delivery in WSNs, provided that new challenging issues related to energy saving and packet forwarding are properly faced. A WSN typically provides information-centric services: in fact, whenever a sensor node is queried, the asking user is interested in the information acquired by the node rather than establishing a point-to-point remote communication. In other words, the user just needs to acquire information about some sensed variable (i.e., the temperature within a room) without caring about the node that will actually provide the asked data.

In addition to that, ICN paradigm distinguish between the routing and the forwarding of the data since they implement new structures charged to realize this in a different manner. Since it is data-based, the routing in ICN depends on the properties of the object name which enables to map directly to the corresponding requested content. In contrary, the data forwarding is based on the ICN architecture that offers tables charged to forward the data from a node to another [10].

Furthermore, ICN paradigms include a potential of in-network caching that makes it a good candidate to be integrated into WSNs. Certainly, in ICN, data can be stored everywhere in the network providing a copy of the same content perceived as a unique content. This represents the most common and important feature of ICN architecture. This feature is considered to alleviate the pressure on the network bandwidth and to improve the content dissemination [11].

It is worth to note that ICN architecture has a lot of potentials that make it a key technology for data dissemination and caching in WSNs. However, its feature has to be adapted to the wireless communication

medium of sensor networks. Several ICN paradigms have been proposed. Nevertheless, in this manuscript, without a lack of generality, we focus on a particular type of ICN paradigm known as Content-Centric Networking (ICN) which was identified as one of the most suitable for WSN [12–14].

1.3 Problem statement

Despite the too many advantages that ICN can bring when enabled in WSNs, it still has several aspects to be studied. Indeed, the ICN features have to be adapted with the sensor network limitations.

In general, IoT applications impose stringent requirements in terms of information freshness since new information is constantly being generated and consumer are mainly interested in the latest information [15]. In a WSN context, when requested by a user, data moves from a node to another and takes time to reach the user. As a consequence, content copies in intermediate nodes may become obsolete. In ICN, when another user asks for the same data, he can get outdated data. However, in most WSN applications, data must be valid to ensure the network reliability. In fact, stale information in the context of fire monitoring in the forest can risk habitant lives. Consequently, it is not only important to check the validity of the data in intermediate nodes but also it is primordial to implement mechanisms that ensure the update of the data when this latter expires.

Besides, in wireless sensor networks, the traditional way of the broadcast is flooding. Then, ICN architecture has to be adapted to work with flooding which is used to disseminate data. Yet, flooding is not considered to be energy efficient because it suffers from data redundancy problem [16]. However, it was demonstrated that broadcast delivery is actively utilized for designing new routing schemes since it seems well fitted to the nature of ICN [17]. Consequently, a forwarding scheme should be found to deal with the broadcast medium of WSN and to minimize energy consumption.

ICN also offers in-network caching that contribute to alleviating the pressure on the network bandwidth in WSNs while spreading content copies between the network nodes in a distributed and efficient manner. Thanks to this feature, users can recover faster the requested content from the intermediate nodes. Then, the traffic load could significantly be reduced and the data availability could increase. However, it is important to ensure that the adopted in-network caching has to be efficient and manage content distribution in an intelligent way. In-network caching in ICN depends essentially on the caching strategy which identifies the content placement and on the cache replacement policy that decides on which content to eject from the

cache once this latter is full. Therefore, these two strategies have to be well investigated in order to fit the WSNs requirements.

1.4 Contributions

In this thesis, our goal is to integrate CCN in WSNs and try to explore its features to tackle the existing limitations in WSNs.

Then, we first focus on content object freshness [18] and on the manner its update is done in the nodes content store. In WSNs, exchanged data must be fresh enough to ensure a certain level of Quality of Service (QoS). Therefore, we design two approaches for the content update in content-centric wireless sensor networks, a reactive and a proactive one to guarantee content validity [19].

The second topic is devoted to forwarding in CCN-WSNs which is achieved via costly broadcasting. In this contribution, we have two main objectives. Firstly, we want to reduce the amount of energy consumed especially when forwarding data. Secondly, we aim to guarantee a high-interest satisfaction rate by proposing a new forwarding scheme. The proposed mechanism depends on a duty-cycle and is applied to the nodes considered as less active (nodes that have a lot of unsatisfied interests in their PIT). Thereafter, we present the impact of the proposed mechanism on energy consumption and network lifetime. To this end, we proposed a mathematical model for energy consumption in CCN-WSNs.

Finally, the last contribution is related to the in network-caching. Our goal is to study the existing caching strategies and try to see the impact of certain parameters by proposing a caching strategy while considering the network constraints and maximizing the network lifetime. To ensure a certain level of diversity and energy efficiency, we propose a caching placement strategy that chooses the nodes on which to cache by taking into account the node degree and its distance from the source node. We also proposed a solution to avoid as possible the problem of interest loop in CCN-WSNs which generates energy waste and latency. In order to evaluate the performance of the proposed strategy, we implement and compared to several existing caching strategies.

1.5 Manuscript outline

The rest of the thesis is organized as follow. Chapter 2 is dedicated to present the background of the thesis. We start by detailing the current state of the art related to the Internet of Things paradigm. We then detail

wireless sensor networks and its integration in IoT. Furthermore, this chapter presents the ICN paradigm, its features, and limitations. It also discusses the adequate choice of architecture for WSNs. Therefore, it cites different existing ICN paradigms, their features and the requirements that should be fulfilled by the adopted ICN architecture.

Chapter 3 studies the suitability of the content-centric networking paradigm for WSNs. Hence, it provides a state of the art of the CCN architecture and the advantage of its integration in WSNs. It specifies the different CCN entities and projects them in the context of sensor networks. Hereafter, it cites works interested in cache freshness, routing and forwarding and in-network caching in CCN paradigm. Finally, it details the Paul Sabatier application scenario in neOCampus project [20].

In chapter 4, we introduce our first contribution related to content freshness in CCN enabled WSNs. Therefore, we focus on the importance of data freshness in the context of wireless sensor networks applications. Afterward, the chapter presents the two proposed approaches for the content update in the content store, the proactive and the reactive one. Then, the chapter gives the simulation results and the difference between the proposed strategies.

Chapter 5 provides a duty-cycling approach for CCN-WSNs. Indeed, in this chapter, we use the concept of duty cycling in WSNs. Then, we start by presenting the different existing definitions of duty-cycle in WSNs and we introduce a taxonomy of existing duty-cycling scheme. Thereafter, the chapter introduces the proposed CCN node model and the mathematical energy consumption model. It then details the proposed duty-cycle scheme in an algorithm for a better explanation. Last but not least, it details the simulation results under two different thresholds and it analyzes the findings for several metrics.

Chapter 6 focuses on our last contribution that concerns the caching decision in CCN enabled WSNs. Therefore, it starts by presenting the motivation of content caching in WSNs. After that, it shows how content is cached in CCN depending on the cache placement and replacement decisions. It also studies the existing caching strategies and tries to see the impact of new parameters on the decision of cache placement and replacement. It explains the different steps of the proposed caching strategy. It finally evaluates the performance of the strategy and compares it to other approaches.

Finally, chapter 7 summarizes this thesis and the work carried out. It reminds the addressed problems, highlights the contributions and summarizes our results. Moreover, it opens future directions to continue working on.

CHAPITRE 2

State of Art

Contents

| | | |
|------------|---|-----------|
| 2.1 | Introduction | 8 |
| 2.2 | Internet of Things | 8 |
| 2.2.1 | Architecture of the Internet of Things | 9 |
| 2.3 | Wireless Sensor Networks | 10 |
| 2.3.1 | Architecture of a sensor node | 10 |
| 2.3.2 | Energy conservation techniques in WSNs | 12 |
| 2.3.3 | Routing protocols in wireless sensor networks | 14 |
| 2.3.4 | WSN as a part of the IoT | 15 |
| 2.3.5 | Challenges of integration of WSNs in IoT | 16 |
| 2.4 | Information-Centric Networking Paradigm | 17 |
| 2.4.1 | The evolution of the Internet and the motivation behind ICN | 17 |
| 2.4.2 | Difference between host-centric and data-centric | 18 |
| 2.4.3 | ICN features | 18 |
| 2.4.4 | ICN limitations | 20 |
| 2.5 | Which ICN paradigm for WSNs? | 21 |
| 2.5.1 | Publish and subscribe | 21 |
| 2.5.2 | NDN | 22 |
| 2.5.3 | CCN | 23 |
| 2.6 | Conclusion | 24 |

2.1 Introduction

The Internet of Things (IoT) is a new paradigm where everyday objects can be equipped with identifying, sensing, networking, and processing capabilities. These capabilities will allow them to communicate with other devices and offer services over the Internet to accomplish some objectives. In IoT, low power integrated circuits, wireless communications, and computation advances ensure low cost, low power, and efficient devices to realize remote applications. The combination of these factors made possible the use of a large number of intelligent sensors and actuators, enabling the collection, processing, analysis, and dissemination of valuable information [21]. Then, Wireless Sensor Networks (WSNs) are considered as an essential element in the IoT.

Wireless sensor networks are among the pioneer networks used for data collection in a variety of applications. Indeed, they are intercepted as information-oriented networks. In such data-driven network, it may seem spontaneous to turn to Information-Centric Networking (ICN) that changes from the traditional host-centric to data-centric communication architecture. Recently, ICN has gained a lot of attention from the wireless sensor network community.

In this chapter, the concept of the Internet of Things is introduced. Then, wireless sensor networks and their integration in IoT are presented. The ICN paradigm and its characteristics are further detailed. Different ICN paradigm that present potential candidate for WSNs is also listed.

2.2 Internet of Things

The Internet of Things aims to connect everyday objects such as watches, cars, smart-phones, etc. IoT is widely regarded as the number one of top 10 technologies that will change the world in the next 10 years [22]. In the traditional Internet, data producers and consumers are human beings. However, in the Internet of things, the main actors become things. Thanks to the recent advances in Radio-Frequency IDentification (RFID), low-cost wireless sensors devices, and web technologies, the IoT has gained a huge interest in connecting everyday objects to the Internet. The Internet of Things is a paradigm where everyday objects can be equipped with identifying, sensing, networking and processing capabilities that will allow them to communicate with one another and with other devices and services over the Internet to accomplish some objectives [1].

Despite the diversity of research on IoT, its definition remains fuzzy. Authors in [23] tried to address too many definitions and architectural models for IoT offered by standardization organizations and IoT projects. According to the IETF, the IoT is an extension of Internet technologies to constrained devices, moving away from proprietary architectures and protocols. As for the IEEE standard association, it defined the IoT as a network of items, each embedded with sensors which are connected to the Internet. In [24], they define IoT as in the context of the Internet, such an addressable and interconnected things that act as the main data producers, as well as the main data consumers. Computers will be able to learn and gain information and knowledge to solve real-world problems directly with the data fed from things. Then, they will be able to sense and react to the real world. The Internet of things, that effectively interconnects a large number of smart devices, creates an environment wherein things have interfaces and identities and can communicate through standard and interoperable communication protocols [24]. In addition to that, IoT combines various network infrastructures such as sensor networks, wireless networks, and the standard Internet, in order to retrieve useful information from things, interact with the physical world, and provide various services in different applications.

2.2.1 Architecture of the Internet of Things

Until now and despite the numerous proposals, there is still no consistent common architecture for the IoT. Too many architectures for IoT were proposed, the basic model is a 3-layer architecture consisting of the Application, Network and Perception Layers [1]. In Fig. 2.1 , we present a protocol stack that covers prominent protocols corresponding to this architecture.

The different architecture layers of IoT are:

- Perception Layer: this layer enables the sampling of the data about the environment like temperature, weight, motion with various kinds of perception devices. It processes data with a cooperative accessing way to obtain useful information and then delivers it to the network layer.
- Network Layer: It permits the fusing of perception information and its transmission to the corresponding platforms of the upper layer via the Internet, wireless network, etc.
- Application and service layer: divided into two sub-layers, service sublayer and application sublayer. Service sub-layer stores and integrates the information from the network layer. It provides much information including information management, data analysis, and decision making. Application

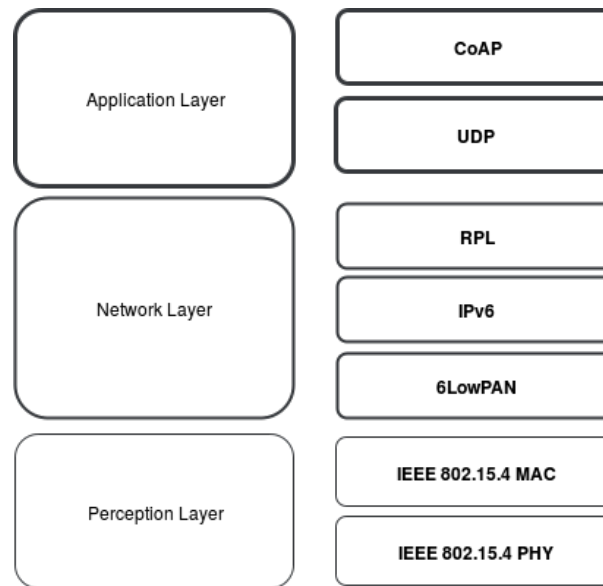


Figure 2.1. An example of an IoT protocol architecture inspired from [1].

sublayer integrates all functions of lower layers and provides service specific for all industries

2.3 Wireless Sensor Networks

Wireless sensor network is a new paradigm in wireless communication networks. WSN has gained lots of attraction both in the academic and industrial sector. A Wireless sensor network is a network composed of a large number of low-power devices that sense the environment and communicate information to one or more sinks [25]. The devices composing a WSN are called sensor nodes or motes, and they have the following characteristics:

- A small size.
- A limited memory and processing power.
- A limited energy (most of them are battery powered).
- Limited capabilities in terms of sensing, data processing, and communication components.

Generally, the WSNs are densely deployed because of these characteristics.

2.3.1 Architecture of a sensor node

A sensor node is composed of four units, each corresponding to a particular task of sensing, processing,

transmission, and power generator [2].

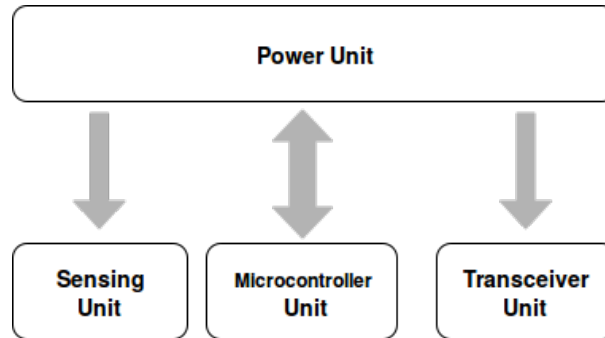


Figure 2.2. Components of a sensor node inspired from [2].

2.3.1.1 Sensing unit

The operating principle of the detectors is often the same: to respond to a variation of the environmental conditions by a variation of certain electrical characteristics, for example, the temperature. The voltage variations are then converted by an Analog-Digital converter so that it can be processed by the processing unit. We can find also MEMS (for MicroElectroMechanical Systems) [2] which are other complex structures to detect other phenomena such as acceleration, chemical concentration, etc.

2.3.1.2 Processing unit

The micro-controllers used in the sensor networks are low energy consumption. Their memory size is about tens of *Kbytes* RAM for data and tens of *Kbytes* ROM for programs [25]. This memory consumes a huge part from the memory reserved for the micro-controller. Generally, the processing memory is associated with a small storage unit in order to manage the procedures that make the sensor node collaborate with the other nodes to carry out the assigned sensing tasks.

2.3.1.3 Transceiver unit

It connects the node to the network. For wireless sensor networks, the transmission consumes about 20 *mW* within few meters. The amount of energy required for transmission increases with distance. To increase these distances while preserving energy, the network uses a multi-hop routing [26]. This disables to turn nodes to sleep mode which consumes more in terms of energy.

2.3.1.4 Power unit

This is one of the most important units of a sensor node. Usually, sensor nodes are battery powered and without possible recharging, their lifetime is limited. Sometimes, power units may be supported by a power scavenging unit such as solar cells [25] which helps in restoring the energy consumed by replenishment. A sensor node has to store the energy and supply it in the required form.

2.3.2 Energy conservation techniques in WSNs

As sensor nodes in WSN have limited battery power, it becomes challenging to perform computation and transmission while optimizing energy consumption [8]. Experimental measurements have shown that generally, data transmission is very expensive in terms of energy consumption, while data processing consumes significantly less. To maximize the lifetime of the network, energy conservation techniques have been developed. Energy-saving techniques focus on two sub-systems: the networking sub-system (the operations of every single node, as well as the design of networking protocols), and the sensing subsystem (the amount or frequency of energy-expensive samples) [27]. Energy efficient protocols are designed to minimize energy consumption and network activity [28]. However, a significant amount of energy is consumed by the components of a node (CPU, radio, etc.), even if they are inactive. Too many techniques exist in the literature that proposes to turn off the components of the node when not necessary. As depicted in Fig. 2.3, energy conservation are divided into three scheme classes: duty-cycling, data-driven, and mobility-based.

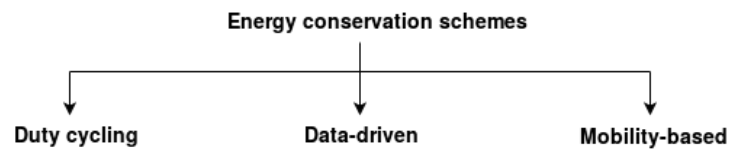


Figure 2.3. Classification of energy conservation schemes.

2.3.2.1 Duty-cycling techniques

Duty cycling can be achieved through two different and complementary approaches by controlling the topology or by managing the power [27]. The most effective way to conserve energy is to put the transmitter radio in standby mode (low-power) whenever communication is not necessary. Nodes that are not currently

needed for ensuring connectivity can go to sleep and save energy. A duty cycle is defined as the fraction of time when the nodes are active. Since sensor nodes perform cooperative tasks, they must coordinate their clocks of sleeping and waking. A Sleep/Wake up scheduling algorithm is part of any plan of Duty-cycling.

2.3.2.2 Data driven techniques

Generally, Duty-cycling plans do not take into account the data collected by the nodes. Therefore, data-driven approaches may be useful for improving energy efficiency. Data-driven techniques are designed to reduce the amount of data sampling by ensuring an acceptable level of accuracy. The reduction of data can be realized by the in-network processing which ensures the data aggregation at the level of the intermediate nodes. A review on in-network processing techniques is presented in [29]. Sometimes, the sending of redundant information to the base station may be unnecessary [30]. Certainly, if it is not used in an intelligent way, redundancy will cause a waste of energy due to redundant transmissions and reception operations. There are several explanations in the literature that find that eliminating any redundancy helps to save a lot of energy and combining them is expected to save more. In [31], we implemented a redundancy-based protocol OER 'Optimization of Energy based on Redundancy' that exploits redundancy for energy efficiency and we enhanced it by proposing a fault-tolerance mechanism.

Besides, a number of data compression methods exist in the literature [32] showed that data compression can be applied also to reduce the amount of information transmitted by the source nodes. This technique involves the encoding of information at the level of the nodes that generate data and the decoding at the level of the sink.

2.3.2.3 Mobility techniques

In [27], they classified mobility-based schemes to mobile-sink and mobile-relay schemes. Mobile nodes can be considered as part of the network or part of the environment. To make nodes mobile, sensors have to be equipped with special components to change their location which may cost a lot since mobilizers are expensive and mobility may result in energy wasting. Hence, mobility may be limited to some nodes that do not have energy constraints comparing to others on the network. Mobility may be also efficient for energy conservation since when sinks are static, traffic is loaded on nodes more than others depending on the network topology and packet generation rates. In the case of the mobile sink, the traffic may be alleviated on different parts of the network and short path may be found when replying with a data which enable energy

saving. An approach exploiting multiple mobile sinks in [33] shows that with this approach the network lifetime could become five/ten times longer than with the static sink approach. In the case of mobile-relay, one of the most well-known approaches is given by the message ferrying scheme. Message ferries [34] ensure moving communication infrastructure to enable data transfer in sparse wireless networks. Static sensor node waits for mobile-relay to pass and send data to it which enables the use of short-range radio signals and hence low energy consumption.

2.3.3 Routing protocols in wireless sensor networks

Various routing protocols are proposed for sensor networks. These protocols either use the flooding or the gossiping concept. The flooding is a method where every packet received is retransmitted to all the nodes in the network. The gossiping tries to enhance flooding by maintaining a probability of transmission. Then, the nodes have a probability p to broadcast the packet they receive. Thereafter, with a probability equal to $(1 - p)$, the packet is discarded [35].

2.3.3.1 Taxonomy of routing protocols

A taxonomy of routing protocols in WSNs is based on various classification criteria such as data-centric, hierarchical, location-based, etc [36] is presented in this section.

Hierarchical Routing In hierarchical routing, sensor nodes are organized into clusters [36]. Based on the energy level of each node, different tasks of sensing or transmitting can be assigned to nodes. For instance, nodes with a high energy level can be chosen as cluster heads. The cluster-head aggregates the data collected from the members of its group and forwards it to the sink [37]. In [8], a distributed clustering algorithm called LEACH for routing in homogeneous sensor networks was proposed. In LEACH '*Low-Energy Adaptive Clustering Hierarchy*' [38], nodes self-configure into groups. Each group is controlled by an elected cluster-head. The cluster-head selection is based on the energy level of each node. In fact, the cluster-head is chosen randomly and its role is assigned to different nodes to ensure fair energy consumption because the cluster-head consumes a lot of energy.

Location Based Routing In location-based routing protocols, the information about the localization of nodes is used for communication. Location routing is based on the position of nodes rather than its network

address. This type of protocols favors energy conservation by calculating the distance between two sensor nodes before transmitting the data. It does not require flooding and hence reduces the control overhead. It only knows the location of its direct neighbors to forward the packet [39]. Geographical and Energy Aware Routing (GEAR) [40], and Geographic Adaptive Fidelity (GAF) [41] are two location-based routing protocols.

QoS based Routing Applications in WSNs require an efficient Quality of Service routing protocols for data delivery from the network infrastructure. QoS based routing ensures reliability and guarantees delays of critical events. Then, QoS based schemes allow sensor nodes to balance between the energy consumption and certain predetermined QoS metrics, such as delay, energy, reliability, bandwidth, etc. before they deliver the data to the sink node [42]. Several QoS based protocols exist such as multi-path routing protocol, SAR protocol [43], energy-aware QoS routing protocol [44], SPEED protocol, etc. Each one is based on a given QoS and tries to ensure a certain level of QoS related to the running application.

Data-Centric Routing Due to the lack of global identification along with random deployment of sensor nodes [36], data-centric routing was proposed. Data-centric protocols focus on the data rather than on the address of the destination. In this routing, an attribute-based naming is necessary to specify the properties of data. Queries in data-centric routing are sent to specific areas and the sensors in that specific area send the data back to the sink. At this level, the first data-centric protocol used is SPIN [45] which checks the negotiation of the data between the nodes, eliminates the redundancy and also saves the consumption of the energy [46].

As an important data-centric routing protocol in wireless sensor networks, Direct Diffusion [47] suggests the use of attribute-value pairs for the desired data and queries sensors on an on-demand basis. In Directed Diffusion, if a sensor wants to receive data, it sends interests for named data. Hence, when data is sent by a source sensor, the data can be cached or transformed by an intermediate sensor, which in turn may initiate interests based on the data that were previously cached [36]. Other data-centric protocols exist in the literature [48].

2.3.4 WSN as a part of the IoT

Wireless Sensor Networks are playing more and more a key role in several application scenarios such as

health-care, agriculture, environmental monitoring, and smart metering. Furthermore, WSNs are characterized by high heterogeneity [49].

Recent technological advances in low power integrated circuits and wireless communications have made available efficient, low cost, low power miniature devices for use in remote sensing applications [21]. Wireless Sensors networks are considered as a promising element of the IoT since they are very compliant to its expectations and this is due to their monitoring and data collecting capacities. The combination of these factors has improved the interest in using a sensor network.

For sensors backbone in IoT, information may be produced either on demand when another sensor requests it (query-based) or proactively sent to multiple subscribers (event-based).

2.3.5 Challenges of integration of WSNs in IoT

When functioning in an IoT network, sensors are assigned additional responsibilities and have to accomplish the following challenges:

- Security: depending on the application sensitivity and in normal cases, sensor nodes have to ensure data confidentiality. By opening WSNs to Internet [50], attackers would be able to threaten WSNs from everywhere. The use of existing security mechanisms already implemented for the Internet is impossible due to the sensor constraints in terms of energy and memory. Then, innovative security mechanisms must be developed in respect to the resource constraints to protect data generated by WSNs from novel attack generating from the Internet [50]. Ensuring security is critical as the system is automatically linked to actuators and protecting the systems from intruders becomes very important [21].
- Quality of Service: sensor network is supposed to contribute to the quality of service by optimizing the resource utilization of all heterogeneous devices that are part of the future Internet [50]. However, the existing approaches ensuring QoS in the Internet are not applicable in WSNs, as sudden changes in the link lead to a change in the configuration of the WSNs topology. New approaches have to be found to ensure delays and overcome losses.
- Scalability: as the IoT will become a globally interconnected infrastructure that includes WSNs, supporting experimentation on an adequate scale is an important aspect. Many architectures that guarantee scalability in IoT were presented. These types of architecture have to be adequate to WSN

to enable the addition of new devices, services, and functions for the customer without affecting the quality of existing services. In addition, scalable systems based on caching mechanisms, parallel computations, and hierarchical architectures are required.

- Resources: devices in IoT should be as small as possible. Current technologies help in reducing the size of a sensor. A direct consequence of physical size is limited resources, energy, capability, and memory. Wireless sensor communication usually expires the energy consumption of an IoT device, so it should be kept to the absolute minimum and adequate mechanisms should be developed to realize this.

In any case, a WSN typically provides information-centric services: in fact, whenever a mote is queried, the asking user is interested in the information acquired by the sensors on top of that mote rather than in establishing a point-to-point remote communication. In other words, the user just needs to acquire information about some sensed variable (i.e., the temperature within a room) without caring about the mote that will actually provide the asked data. In addition to that, as mentioned before, IoT proposes to merge sensor networks with the Internet. Then, the environment monitoring applications need the transmission of the data collected from the sensor nodes so that it is further analyzed. Consequently, data-centric protocols seem to be the best candidate for merging WSNs in the Internet of Things.

2.4 Information-Centric Networking Paradigm

The ICN is a data-centric paradigm appeared to face the new Internet requirements. Early publish/subscribe systems can be considered as the first step toward the information-centric networking model.

2.4.1 The evolution of the Internet and the motivation behind ICN

Back to several decades, the connection-centric paradigm met the requirements of networking at that time since the only purpose is to deliver packets between two endpoints. Nowadays, the Internet has been transformed from an academic network to a global infrastructure for the massive distribution of information. Users are interested in receiving information wherever the information may be located, rather than in accessing to the location. ICN is a result of paradigm shifting from host-to-host model to content-oriented model.

Motivation from ICN is a change in user requirements from resource sharing to information dissemination [51]. Indeed, ICN goal is to enhance the data dissemination efficiency in the network to adapt it to

the new requirements of today Internet. To this end, we chose to work with this paradigm. This section introduces the paradigm as well as the advantages that it brings in terms of naming, routing, in-network caching, mobility, and security. Besides, the limitations of this paradigms are addressed further.

2.4.2 Difference between host-centric and data-centric

For the host-centric protocols, each source independently sends data along the shortest path based on the path that the queries took ('end to end routing') [52]. However, for data-centric protocols, the sources send data to the sink based on the content and perform aggregation on the data originating at multiple sources [52]. Therefore, the difference exists in the manner the data is sent.

In [52], they showed that data-centric routing offers significant performance gains across a wide range of operational scenarios. In addition to that, data-centric technologies perform in-network aggregation of data to realize an energy-efficient transmission.

2.4.3 ICN features

ICN presents a lot of features that make it a good candidate to explore in WSNs. These features are detailed hereafter.

2.4.3.1 Naming

The ICN approach fundamentally decouples information from its sources, by means of a clear location-identity split [53]. In ICN contents are identified by persistent and unique names. Using a unique name will generate information under the same name, although the data values will be different. In addition, in ICN, users request a content by its name instead of its location address. A very wide debate is in progress within the research community about the design of content names: they can be either flat such as in MF [54] and XIA [55] or hierarchical as in NDN [56] and CCN [57]. The hierarchical names have a structure similar to URLs. A content may be divided into many chunks. The decision whether to use flat or hierarchical names either human-readable or self-certifying mainly impacts the scalability of the ICN routing plane [58]. In this thesis, we do not treat the naming in ICN.

2.4.3.2 Routing

The shift towards content-centric bandwidth-demanding applications requires the Internet to efficiently deliver massive amounts of information and handle large spikes or surges in traffic, commonly referred to

as flash crowds [53]. In ICN the network may satisfy an information request not only through locating the original information source but also by using (possibly multiple) in-network caches that hold copies of the desired information. Furthermore, in ICN, routing answers to the question *'how interests are routed from users to source nodes that contain the corresponding requested content and then how contents are routed back to the user?'* ICN proposes two routing techniques, name resolution routing, and name-based routing. Unlike name resolution routing, name-based routing is based on the name hierarchy to route request and directly forward them to producers without forwarding them to locators in order to solve the names.

2.4.3.3 Caching

In-network caching is one of the inherent features in ICN paradigm treated in this thesis that facilitates information dissemination. Thanks to the in-network caching offered by ICN, the recovering of the content object is faster and decreases the overhead. Moreover, by supporting in-network caching, ICN avoids repeated delivery of the same content in the network. Along with forwarding content, nodes in ICN can cache content too if the content is not already stored in their cache. Hence, in-network caching was introduced to alleviate the pressure on the network bandwidth and improve the transmission efficiency in content dissemination [59]. ICN in-network caching is ubiquitous since any ICN node can be a cache [59]. In the Internet of Things, objects have some constraints related to their memory i.e. their caching storage is limited. Hence, once the cache is full, a caching replacement policy is performed.

2.4.3.4 Mobility

The ICN paradigm does not have end-to-end connections so it does not expose the problem of managing this type of connection [60]. A mobile user just continues sending an interest for a defined content object and different source nodes may respond with the content. Then, there is no need to maintain a connection with the previous source. Even when content changes their location, they will be always reachable since they do not depend on the location. Consequently, mobility in ICN can be consumer mobility or content mobility. When mobility occurs, the routing information must be updated in all the nodes belonging to the request path [58].

2.4.3.5 Security

ICN architectures are interest-driven, i.e., there is no data flow unless a user has explicitly asked for a particular piece of information. This is expected to significantly reduce the number of unnecessary data

transfers and also to facilitate the deployment of accountability and forensic mechanisms on the network points that handle availability and interest signaling. Moreover, for ICN architectures the use of self-certifying names for information, malicious data filtering will be possible even by in-network mechanisms. Finally, most ICN architectures add a point of indirection between users requesting a piece of information and users possessing this piece of information, decoupling the communication between these parties. This decoupling can be a step towards overcoming denial of service attacks, as requests can be evaluated at the indirection point, prior to arriving at their final destination. Indirection can also benefit user privacy, as a publisher does not need to be aware of the identities of its subscribers [53].

2.4.4 ICN limitations

Although ICN offers new features, most of the ICN designs provide only basic instructions, which opens a large research field. The contributions in this area mainly focus on enhancing the routing and forwarding of information, caching management and security of contents.

2.4.4.1 Caching

Cache management depends on the cache decision and replacement policies. The node decides which content should be cached when a new data object is received, and it selects the content should be evicted when the cache is full [61]. The capabilities of caching mechanisms affect the availability parameter in different scenarios [62]. Several issues like overflow memory and timing update are faced while caching. Hence, appropriate caching strategies should be proposed to overcome these problems and to ensure the purpose of availability of data or path towards data.

2.4.4.2 Routing and Forwarding

The usage of multiple-source multiple-destination presents challenges in terms of information-forwarding in ICN [62]. Few implementations of ICN support this feature. The forwarding management is ensured via the implementation paradigms like ICN. The naming scheme used in ICN could also present some challenges for routing. Certainly, it has to be adapted to the routing protocol and to the use case. However, forwarding strategies should be enhanced or proposed to ensure the caching provision and efficient content delivery.

2.4.4.3 Congestion control

In ICN architectures, multiple requests and responses are generated from multiple sources. This behavior

may result in huge delays, collisions and packet loss. Even if multiple congestion control mechanisms have been proposed for ICN such as Receiver-driven TCP-Reno congestion control [62], stronger mechanisms have to be defined.

2.4.4.4 Security

Unlike nowadays host-centric security (e.g., transport layer security (TLS), authenticated server), ICN requires location independent security mechanism to enable ubiquitous in-network caching system [61]. In ICN, data is broken down into independent content chunks that can be sent or received. Usually, these chunks are encrypted but still some important information as content name, timing and size can be accessed [62]. So, there is a need to define appropriate mechanisms for security and privacy. Therefore, the next generation security model should provide an information-oriented data integrity and authenticity check mechanism [61]. In this thesis, we do not consider security issues.

Besides to those major directions in the optimization of ICN feature performances, another topic of research is to examine its feasibility in real-world ICN implementation.

2.5 Which ICN paradigm for WSNs?

Since sensor networks already provide data-centric services and since information-centric networking became one of the significant directions nowadays, it is challenging to see if the attractive features offered by this paradigm can fit WSNs and overcome some existing limitations. A certain compliance between ICN and WSNs have to be ensured.

Different Information-Centric Networking architectures have emerged during the last decade. Most of them share the same characteristics. The most common point in this architecture is changing content addressing by its name. Moreover, caching and content awareness has a key property in ICN. In this section, we list some general ICN paradigms and we make a decision on the one that we are going to adopt for our contributions during this thesis.

2.5.1 Publish and subscribe

Publish/Subscribe usually relies on a set of intermediate nodes and brokers that gather and dispatch all the packets sending [63]. Brokers need to know nodes identities and to maintain long-range routes. In

WSNs, computing power, memory, and energy are scarce resources. Publish/subscribe appears as an efficient communication scheme for WSN by gathering all the information from the sensor nodes (publishers), treat them, and transmit the resulting data to the collection node (subscribers) [63]. Inside the brokering system, any data manipulation algorithm can be applied (aggregation, fusion, filtering, encryption, etc.), therefore the flow conveyed to the subscribers is not necessarily the sum of the incoming flows. All communications pass through the brokers, so it depends on the number and the location of brokers in the network. Data should take the shortest path from the subscriber to the broker and from the broker to the subscriber [63]. In Wireless Sensor Networks, all data is broadcasted and filtered based on MAC address. Thus, energy consumption in a node does not only depend on the number of the frame it forwards but also on the number of frames sent in its neighborhood. Brokers should be distributed and located in a central area. The functionalities of the publish and subscribe paradigm are shown in Fig. 2.4.

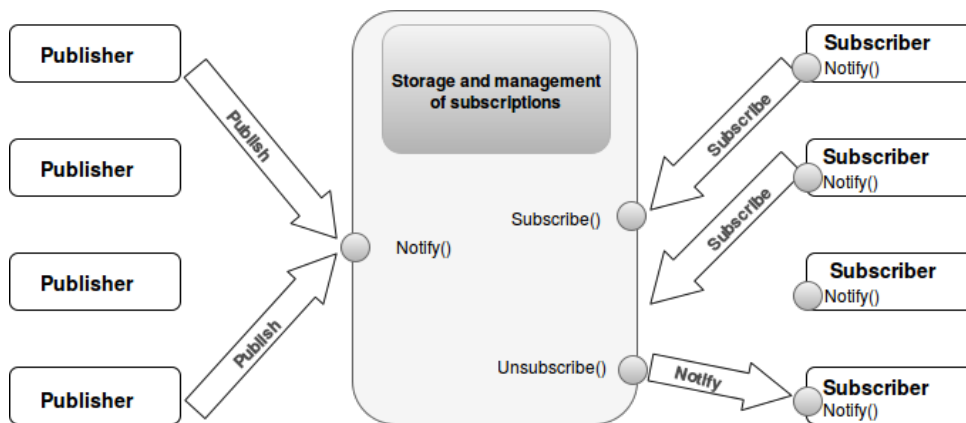


Figure 2.4. Publish and subscribe paradigm [3].

2.5.2 NDN

Named Data Networking (NDN) is one of the Information-Centric Networking architectures for the future Internet that is funded by the National Science Foundation (NSF). NDN uses data names to communicate contents in a network rather than node addresses. Therefore, this network architecture for future Internet does no longer concentrate on 'WHERE' the information is located, but on 'WHAT' information is needed. Hence, this paradigm shifts from a host-centric to data-centric and creates new challenges to efficiently

exchange data between data consumer and provider. NDN communication is initiated by the consumer through the sending of an Interest packet. Once the interest reaches a publisher or a node having the corresponding content object packet, it replies. Each NDN structure maintains a Content Store (CS) to cache contents; a Forwarding Interest Base (FIB) to store forwarded interests and a Pending Interest Table (PIT) to record unsatisfied interests. The forwarding process of Interests and Content Objects is illustrated in Fig. 2.5.

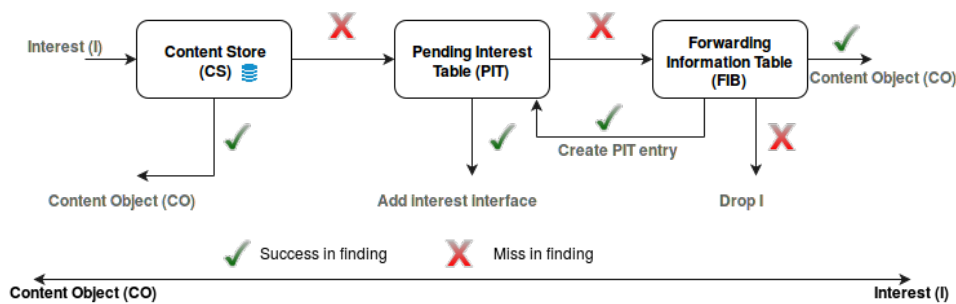


Figure 2.5. NDN/CCN Information routing.

Therefore, when an NDN router receives an interest, it checks its CS; if the content is available, it replies with it. Otherwise, it looks to its PIT, if the interest is available, it discards the Interest. If not, it checks its FIB, if the interest exists, it adds an entry in its PIT and it transmits the interest to look for it in the path indicated in the FIB and it replies with it. If not found in FIB, the interest is discarded.

NDN routing protocol coordinates with NDN forwarding plane for interface ranking and probing, the only difference between routing and forwarding is that, while routing decides about the availability of routes, forwarding makes decisions about the preference and usage of routes based on their status [64]. There are mainly three major routing protocols in NDN listed in [65–67].

Moreover, NDN opens new perspectives in the way data can be retrieved in wireless sensor networks since its features match the use cases and applications developed on sensors.

2.5.3 CCN

The Content-Centric Networking (CCN) [68] architecture is an ICN implementation from Palo Alto Research Center (PARC). CCN is also built on named data where the content name replaces the location address. Its

services include multihop forwarding, flow control, and loop-free multipath forwarding. Thus, CCN delivers named content to the user from the nearest cache or content provider. Applications use the CCN on top of lower layer communication services that can handle packet transmitting [13]. This protocol is flexible and can be deployed in different environments where providing data content is an important concern. CCN uses hierarchical naming of content that looks like the URL scheme of today Internet. These names are prefixes that match to existing contents. For instance, a student asking for the temperature of the classroom 204 in the Building U4 at Paul Sabatier University. He sends an interest: /Collect/TemperatureBuildingU4Classroom204 and the node with the corresponding content replies. The forwarding in CCN is ensured via interest and content object packets as it was realized in NDN in Fig. 2.5. The original CCN [69] was the first query-based ICN architecture applicable to wired networks.

Other ICN architectures exist in the literature such as Network of Information (NetInf), the EU funded project named 4WARD/SAIL, the Publish-Subscribe Internet Routing Paradigm (PSIRP), and the Data-Oriented Network Architecture (DONA) by UC Berkeley etc.

Among several ICN architectures discussed previously, the Content-Centric Networking is considered as a suitable candidate for WSNs. The CCN includes CCNx [68], a protocol which is built on named-data and provides location-independent delivery. It includes multi-hop forwarding for end-to-end delivery, flow control, transparent multi-cast, loop-free multi-path forwarding and verification of content integrity [70]. Besides, it supports a wide range of networked applications and can be deployed in different environment mainly where providing data is the most important concern like in WSNs. By dint of all these reasons, we decided to try to enable CCN in WSNs to take benefit of its features.

To introduce our work, we focus on content-centric networking but we argue that our proposed solutions can be applied (with some modifications) to any ICN architecture that works as a network of caches in pull mode. Wireless sensor networks present a lot of challenges when applying this type of architecture. In this thesis, we try to resolve some problems in WSN by taking benefits from the content-centric networking. Besides, we consider some limitations of CCN and we try to enhance them by the proposed solutions.

2.6 Conclusion

By bringing information into the network layer, the ICN approach promises to enhance communication

in WSNs. Sensor networks present limitations in terms of energy, memory and computation capabilities. Besides, the deployment of WSNs and their management present a challenge in terms of coordination between sensors and resource management. However, since WSNs are already information-centric, we argue that enabling the use of ICN in is a good idea. ICN with the features it offers proposes to overcome these constraints. Indeed, ICN in-network caching and forwarding scheme could bring efficient solutions.

Chapter 2 was dedicated to enlightening the general concepts of this thesis, notably the Internet of things, the Wireless Sensor Networks and the Information-Centric Networking paradigm. We presented the different paradigms that exist in ICN and we described their functionalities. In the next chapter, we detail CCN architecture that we consider suitable for sensor networks and we will present the related research close to the topics that we are going to handle in this thesis.

CHAPITRE 3

Content-Centric Approach for Wireless Sensor Networks: Suitability?

Contents

| | | |
|------------|--|-----------|
| 3.1 | Introduction | 28 |
| 3.2 | Content-centric networking approach | 28 |
| 3.3 | Content-centric networking in wireless sensor networks | 30 |
| 3.3.1 | Related work | 30 |
| 3.3.2 | Node structure | 31 |
| 3.3.3 | CCN entities | 33 |
| 3.3.4 | CCN stack in WSNs | 33 |
| 3.3.5 | Advantages of CCN integration in WSNs | 34 |
| 3.3.6 | Content-Centric integration in Wireless Sensor network | 36 |
| 3.4 | Cache freshness | 37 |
| 3.5 | Routing and Forwarding | 38 |
| 3.6 | In-network caching | 40 |
| 3.7 | Application scenario: traffic on Paul Sabatier University | 42 |
| 3.8 | Conclusion | 44 |

3.1 Introduction

Wireless Sensor Networks are an important technology for large-scale monitoring application that provides measurements. WSNs can perform in-network processing operations such as aggregation, event detection, or actuation. In WSNs, usage models are similar to where users are interested in data measured by sensors rather than in sensor nodes. Sensor networks present too many challenges in terms of caching, self-adaptation, as well as security and these challenges are far from being solved.

Content-Centric Networking is a promising data-centric protocol (an ICN paradigm) that offers content caching, self-adaptation, and built-in security at the data level. Approaches like in-network caching and data aggregation are important in WSNs. Therefore, CCN is sawed as a suitable approach to be applied in WSNs to overcome some challenges given by this type of environment.

Combining CCN and WSNs is a new emerging trend to implement real data-centric applications. In CCN, the forwarding of interests and content objects is performed based on a distributed way which normally meets the requirements of WSNs and IoT. Therefore, CCN in wireless sensor networks presents a promised technique that may ensure data routing based on content.

In this chapter, we start by defining CCN and its main features. Then, we introduce the related works of ICN applied in wireless sensor networks. After that, we present CCN architecture applied to WSNs and we list the motivations behind its integration. We also present an application scenario of the Paul Sabatier Campus. In the last part of the chapter, we are particularly interested in discussing the existing works concerning the data freshness, the forwarding strategies and the in-networking caching in CCN.

3.2 Content-centric networking approach

Content-Centric Networking is a new communication architecture with a different model of forwarding that does not talk directly maps to things since there is no host abstraction [71].

CCN is a shift from a host-centric view of the network to a content-centric one. Hence, the focus is not anymore on ‘WHO’ to communicate with like in but on ‘WHAT’ to communicate. The architecture of CCN is detailed in Fig. 3.1.

In CCN, nodes deal with two kinds of packets, an interest packet which contains the name and a few other fields, and the data packet that contains the content. An interest is asking a question like ‘Does any

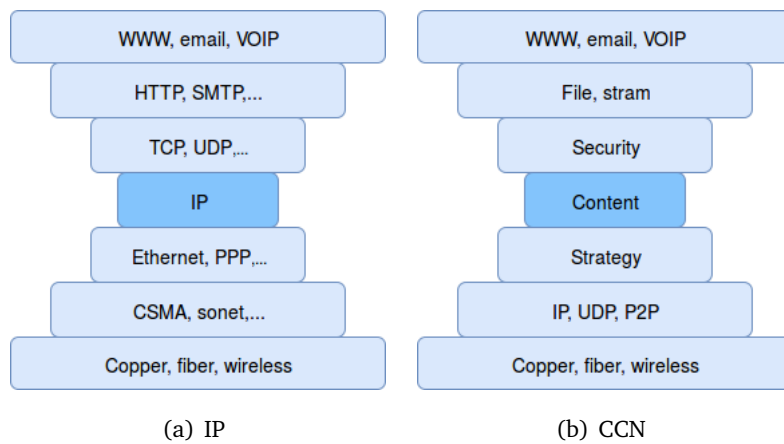


Figure 3.1. IP/CCN stack inspired from [4].

node have any content that matches with this name?’ The answer to this interest is the data. Besides, since ICN paradigms secure the data, the content contains the name, the signature, and some other fields. CCN uses long names derived from human, readable names, and they contain content identifiers at the end of them [71] as shown in Fig. 3.2. The basic mechanism of CCN is the dissemination, a broadcast model. The user sends the interest in the network and the node that hears the interest can respond if it has the content. Otherwise, the interest is deleted.

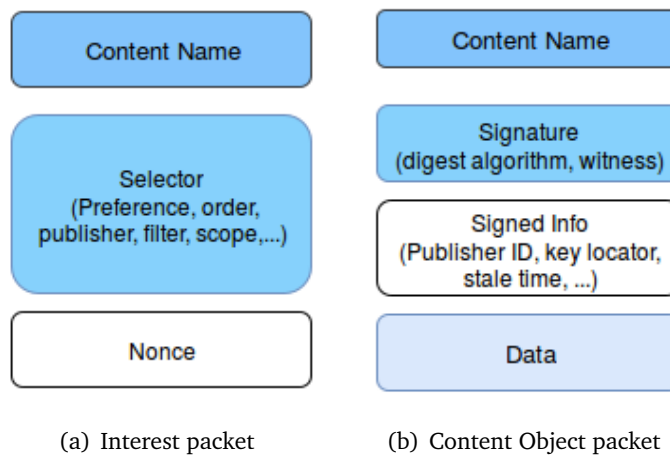


Figure 3.2. CCN packets inspired from [4].

3.3 Content-centric networking in wireless sensor networks

The integration of CCN in wireless sensor networks was and still remain the subject of several research projects. CCN offers features that may help to overcome several challenges in WSNs. Consequently, in this section, we list some existing works in the literature and we present the structure of a CCN node but this time in a wireless sensor network environment.

3.3.1 Related work

Recently there has been a new emerging trend in integrating content-centric networking or named data networking with wireless sensor networks to implement real data-centric Internet of Things. As stated earlier, Named Data Networking or NDN is another project that has its roots in CCN and shares with it almost the same characteristics. Research about implementing this is still in its infancy, however too many works that integrate CCN in WSNs were proposed and validated by simulations.

For instance, in [14], authors proposed to integrate CCN in Wireless Multimedia Sensor Networks (WMSNs). They consider the problem of flooding in Wireless Sensor Networks which is achieved via costly broadcasting. To mitigate this problem, they proposed a new protocol named packet diffusion-limited protocol for CCN based WSNs for smart cities. They showed that the new protocol may suppress data packet flooding and speed up data packet forwarding: so it is a good candidate to be implemented on the top of WMSNs.

As for [10], the authors discussed that integrating NDN in WSNs enables the implementation of flexible forwarding scheme that solves the problem of flooding in WSNs. For this, they proposed DMIF, a 'Dual Mode Interest Forwarding scheme' in which energy efficient mechanisms including flexible mode shift, flooding scope, broadcast storm avoidance, and packet suppression were proposed. Simulation experiments validate the ability of CCN to enhance energy efficiency and network lifetime in WSNs.

Furthermore, the work of [72] introduced the application field, challenges, and concept of information-centric networking approach as a fundamental driver for Wireless Sensor and Actor Networks (WSANs). They showed that exploring ICN in WSANs enables efficient coordination, interoperability, service discovery, and prioritized routing which improves its performances. They implemented a simple test-bed based on information naming scheme to demonstrate the feasibility of this approach.

Yet, in [12], Zhong *et al.*, proposed CCN-WSN ‘a lightweight, flexible Content-Centric Networking protocol for Wireless Sensor Networks’ in which they integrate concepts of CCNx but multiple aspects are modified to meet the WSNs constraints. They modified the message format and proposed a flexible naming strategy to extend the functionality of content names. They showed that the content-centric designed for the Internet is suited for usage in wireless sensor networks in a lightweight variant due to the stringent usage of interest, content and implicit routing.

Concerning researchers in [73], they explored NDN potentialities in WSNs. In this work, the authors showed that NDN features match the use cases and applications developed on top of sensors and well cope with their potential constraints. They also enhanced NDN with packet overhearing to reduce collisions and duplicated transmissions. Besides, they extended it by principles inspired by the data-centric directed diffusion routing technique. They showed that NDN is a promising candidate technology for WSNs.

Last but not least, authors in [74] proposed a two-tier CCN architecture to manage the heterogeneity of devices in WSNs. They showed that CCN provides a complete communication framework for data retrieval and dissemination. CCN offers efficient naming schemes, security mechanisms and novel routing strategy that matches with the requirements of WSNs. They agree that WSNs must be connected to the Internet through which monitoring entities can reach sensor nodes [74]. They also implemented a CCN standard devices such as servers and gateways and a lightweight version directly on the top of these constrained devices.

In Table 3.1, we list different works that enable CCN or NDN in WSNs and we focus on the features addressed in each research.

3.3.2 Node structure

In each node, as presented in fig. 3.3, some data structures are maintained to properly forward the interest in the network [79]:

- *Interface*: An interface is configured to send and receive a broadcast packet. A sensor node only has one interface.
- *Content Store (CS)*: The Content Store is a buffer memory or cache where data is stored. CS is not a persistent store, it holds content created locally or content object received from other nodes in the network. When finding a match in the CS, the processing stops and the interest message is discarded.

| | Naming | Caching | Routing and forwarding | Energy efficiency |
|-------------------------------|--------|---------|------------------------|-------------------|
| Chauchan <i>et al.</i> , [75] | | ✓ | | ✓ |
| Dinh <i>et al.</i> , [72] | ✓ | | | |
| Zhang <i>et al.</i> , [12] | ✓ | | | |
| Amadeo <i>et al.</i> , [73] | ✓ | | ✓ | |
| Jan <i>et al.</i> , [74] | | | | |
| Gao <i>et al.</i> , [10] | | | ✓ | ✓ |
| Gayathri <i>et al.</i> , [76] | | ✓ | | |
| Aboud <i>et al.</i> , [77] | | | ✓ | ✓ |
| Chen <i>et al.</i> , [78] | | ✓ | | |

Table 3.1. Studied CCN features in several CCN enabled WSNs works.

- *Forwarding Information Base (FIB)*: It is similar to the routing table in IP. If the node does not find a matching content in the CS it moves to check the FIB. FIB gives the interface on which the interest should be sent to retrieve a matching data. If finding a match in FIB, an entry is created in the PIT and the message is transmitted to the destination.
- *Pending Interest Table (PIT)*: PIT stores unsatisfied forwarded interests and contains a list of interfaces from which these interests have been received.

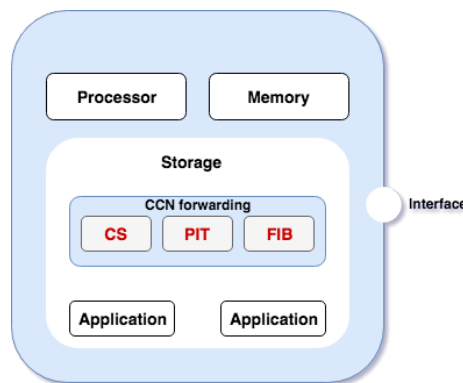


Figure 3.3. CCN node structure elaborated for a sensor node and inspired from [5].

3.3.3 CCN entities

In CCN for WSNs, the communication is based on two types of message: interest and content object. Thus, when a node requests a content since the medium is wireless, it broadcasts an interest and the node having the corresponding content responds with it as depicted in Fig. 3.4.

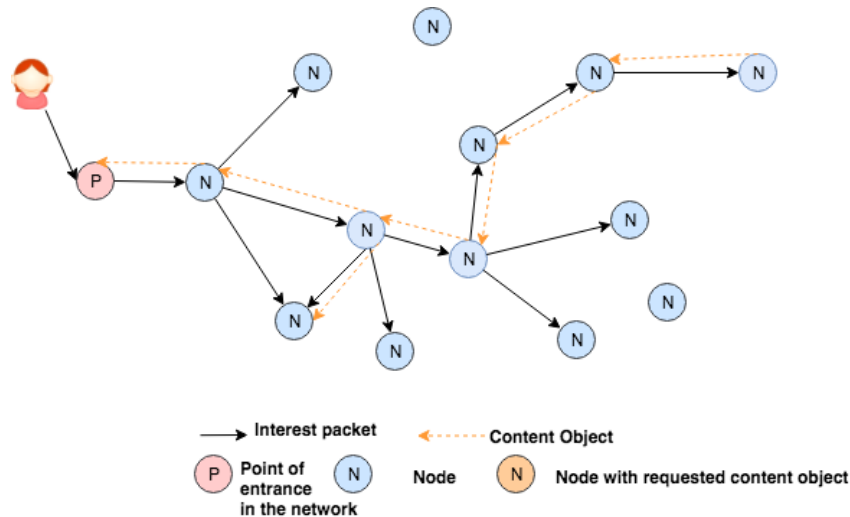


Figure 3.4. CCN operations, interest dissemination and content object delivery.

Therefore, the user interested in receiving a content sends an interest. Once the interest is received by a node in the network, it starts by checking its content store. If a matching happens, it replies with the content. Otherwise, it checks its PIT. If the interest matches with an entry, the interest is discarded. Or then, the node looks in its FIB, if it finds an entry (the content exist on the path), it broadcasts the interest on its interface and the node with the content object replies. In a different way, the interest is discarded (which means that it does not exist in the neighboring). The intermediate nodes keep a copy of the content in their CS if the interest goes through them.

3.3.4 CCN stack in WSNs

As already mentioned, CCN replaces the host-based notion in networks with data-based one. CCN is different from host-centric architectures because it differently names and values information. While IP requires that each packet has a destination and source label associated with it, CCN binds a name to the data itself, allowing the packets to be location-independent. An example of its integration in the wireless sensor

networks stack is described in Fig. 3.5.

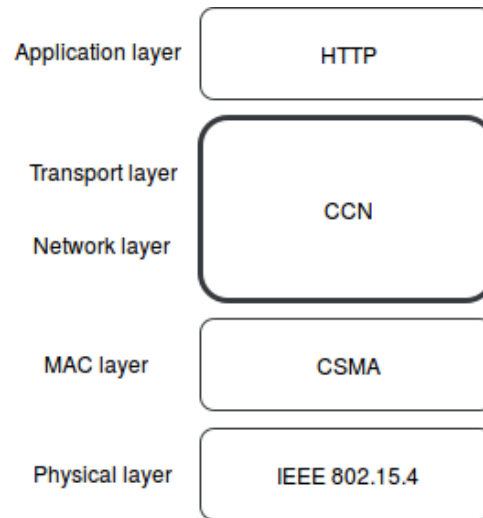


Figure 3.5. An example of a CCN stack in WSNs.

The CCN communication layer handles packet transmission and does not rely on other transport protocols to deliver messages. The communication model is built entirely and uniquely on named data.

3.3.5 Advantages of CCN integration in WSNs

By bringing information into the network layer, the CCN approach promises to provide many benefits for an efficient communication, which is also desirable in wireless sensor networks [72].

3.3.5.1 Content caching

In WSNs, CCN provides caching to reduce congestion and improve the end-to-end delay. Caching ensures high content availability, network traffic reduction, and low retrieval latency. Caching data may ensure store-carry-and-forward communications. CCN may cache both interest and content object. Along the path from the requester to the sender, any cached copy can be used. A node in the network may keep a copy of an interest in the PIT, so it will be able to respond rapidly to the requester if it does not have the data. Data can also be found by local search in the Content Store. In a CCN architecture where location has no value regarding the content, this kind of content awareness combined with caching could reduce the amount of traffic significantly [80].

3.3.5.2 Naming strategy

In CCN, the communication is pull-based, content is sent in response to an interest. WSNs require both pull-based and push-based communication. Besides, the data transmitted in a WSN could be small since they are memory-constrained. Then CCN proposes a flexible naming strategy for CCN-WSN to integrate arbitrary data in the interest message as a name component [12]. Hierarchical naming facilitates content search and retrieval in large-scale WSNs and could make data aggregation easier [12]. Naming is not treated in this thesis.

3.3.5.3 Scalability

CCN is particularly indicated to retrieve data from several nodes in a monitored Sensor Networks area. When the network density increases, CCN may prevent from degradation of the communication quality by controlling forwarding mechanism [73].

3.3.5.4 Mobility

Applications in WSNs monitor animals, people, and mobile devices. In the IP approach, an address indicates the location and the identity of an endpoint, causing many of the mobility issues of end-to-end connection maintenance [72]. Whereas, in Content-Centric Networking, the routing is realized based directly on the name of the content to find the nearest data instead of looking for it in a specific location that may be far from the requester. CCN natively provides sufficient ways to handle node mobility in WSNs.

3.3.5.5 Forwarding

In CCN, a content should be forwarded in response to a previous interest and if there is no content for the interest, it is discarded. The structure of a node in CCN for sensor networks makes easier the interest forwarding and content object. As stated previously, a node contains three tables that enable it to forward interests according to a special forwarding strategy. The node starts by checking its CS, if it finds a content that matches the interest, it responds with the content. When a node finds an entry in its FIB table, interest is flooded in the network until it is responded with a matching content. Otherwise, if there is an entry in the PIT table, interest is discarded. This makes the response faster and avoids the end-to-end path to look for a content that the node may not find at the end [60].

3.3.5.6 Security

Nowadays, wireless sensor networks are completely separated. CCN security mechanism may protect the information from this by securing the data and not the link. Security in CCN must be embedded in the content object [60]. This has been implemented in most of CCN architecture, all the public contents are authenticated with a digital signature and private contents are encrypted with an encryption key. However, security still remains an open issue that we do not consider in this thesis.

The advantages cited previously motivated us to enable CCN in WSNs. Besides, CCN may enhance the forwarding in the content in WSNs by replying faster with the content and avoiding the end-to-end path. Then, our goal by applying CCN is to reduce the energy consumption of the nodes without affecting the interest satisfaction rate. This means without reducing the amount of satisfied interest sent by users. Moreover, the characteristics of CCN, such as query/response communication and in-network caching, may be exploited in WSNs.

3.3.6 Content-Centric integration in Wireless Sensor network

Connecting IoT and WSNs enable the interconnection and the communication of different heterogeneous devices [81]. In our case, the devices are equipped with sensors enabling content-centric protocol to exchange information. Users also use CCN to request contents as shown in Fig. 3.6.

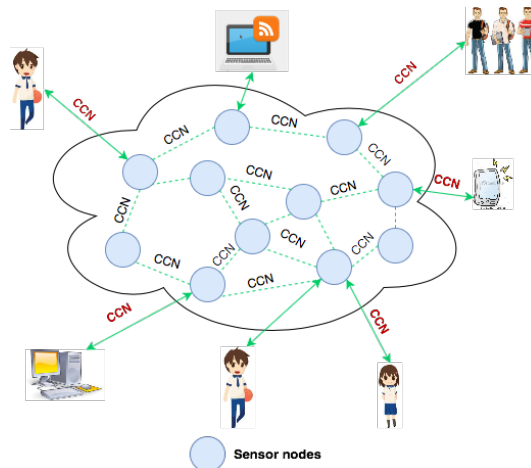


Figure 3.6. CCN enabled WSN communication.

The sensor networks are composed of different smart devices communicating using CCN paradigm. While the principle of CCN is a suitable approach for WSNs, some modifications are done in the CCN protocol to meet the requirements of WSNs. Then, a lightweight and flexible CCN version is implemented [12] in order to overcome memory and computational constraints. Messages format in CCN for WSN is different from messages in traditional CCN version. The most essential fields which are the name and the data are kept in both but some selectors are modified. The node structure is the same, just the concept of faces became just one interface in WSNs as the medium is naturally broadcasting.

3.4 Cache freshness

The cache consistency is an important issue in CCN, which refers to whether cached contents in routers are outdated. This issue is still not investigated thoroughly in ICN [59]. Cache freshness maintains the validity of the shared contents stored in multiple caches. The freshness differs from a data to another, for example, a temperature is considered valid for one hour. A copy of the data is considered valid if it has the same value as the source.

Quevedo *et al.*, in [82], analyze the impact of the inherent multi-node caching mechanism of Information-Centric Networking approaches considering the delay between the information generation and its consumption from the cache. It also proposes a novel consumer-driven information freshness mechanism to overcome the negative effect of the information freshness requirements in IoT-enabled ICN scenarios.

Authors in [59] proposed PCC ‘a cost-effective Popularity-based Cache Consistency’ mechanism to guarantee the freshness of cached contents in ICN routers. This mechanism is able to balance the trade between the consistency strength and related costs since it only maintains the strong consistency for popular contents and weak consistency for unpopular ones.

In [58], the authors designed a freshness mechanism called event-based freshness in IoT. The role of this coherence mechanism is to check the freshness of a requested data found in a cache node before to send it back to the consumer. It depends on producer behaviors, the time of storing a content in a cache called cache-time, the period in the case of periodic mode, and past events in the case of On/Off mode.

In wireless sensor networks, continuous monitoring applications require periodic refreshed data at the sink node. Data reaching the sink node after a certain threshold is not used for processing or analysis

because it is considered stale. Data freshness along with reliable data delivery is critical in such applications. Data freshness can also be improved by minimizing the data propagation in sensor networks.

The cache freshness is an important challenge that threatens the routing and in-network caching in CCN, especially with a high-dynamic environment in term of content updates [58]. Moreover, in wireless sensor networks, data exchanged between sensors have to be fresh to ensure a minimum level of reliability then a certain level of QoS. This issue, despite its importance, is still in its infancy. In this thesis, we are going to treat the problem of cache freshness in the case of CCN enabled WSNs.

The cache freshness may present a condition to realize a cache replacement in CCN for WSN. Once the cache size is full, some invalid cached contents must be evicted from the CS in order to replace it with fresh content.

3.5 Routing and Forwarding

Routing schemes in CCN and IP have the same semantic. CCN nodes have FIBs which contain content names instead of IP addresses. FIBs propose multiple interfaces for a given entry. In WSNs, a unique interface is assigned for all the entries since we are in a wireless network. Forwarding tables are populated by disseminating topology information. In CCN, in addition to forwarding requests, the forwarding plane is responsible for detecting failures and recovering them.

Routing is one of the most important issues to be considered in wireless sensor networks which should balance the energy in the network, increase the packet delivery ratio and maintains the route reliability by adding the self-healing features [83]. Therefore, it is a big challenge to provide an efficient and reliable communication paradigm for WSN data transmission.

A great number of works addressed routing in wireless sensor networks. The authors in [84] proposed a novel flooding design which solves problems caused by low duty-cycle operations and ACK implosion. They mathematically proved that the energy consumption of broadcasting can be reduced by allowing nodes with high correlations to wake up at the same time. Then, they considered a low duty-cycle and they proposed an adaptive flooding scheme. However, in CCN for WSNs, this cannot be realized. Since in CCN, nodes have to ensure a minimum activity cycle. So, the nodes have not to turn on mostly-off mode. In [83], the authors presented a dynamic data forwarding scheme which maximizes the lifetime of a sensor network

by predicting the mobility of the sensor nodes. The scheme operates in two phases, cluster formation, and data forwarding.

In [85], the authors design a one-hop forwarding and routing protocol for multisource data retrieval in NDN based IoT. An interest flooding scheme and a reactive optimistic name-based routing scheme are proposed in [86]. However, these proposed routing schemes do not take into consideration the controlled flooding scope and energy awareness, so that they are not suitable for WSNs with severely limited energy and computation resources.

Regarding routing and forwarding in CCN enabled WSNs, there have been few works in this area. In the literature, Gao *et al.*, [10] focused on interest forwarding and routing issues for NDN based WSNs. Hence, they proposed a Dual Mode Interest Forwarding scheme for NDN based WSNs called DMIF. In WSNs, the request is flooded from a sensor to another until arriving at a node with the corresponding content. The protocol combines two forwarding protocols in which several energy mechanisms including flexible mode shift, flooding scope control, broadcast storm avoidance, packet suppression, and energy weight factors are designed to save and balance the energy consumption. Then, the protocol supports controlled flooding and reduces the network overhead. Besides, in [77] authors proposed a Geographic Interest Forwarding scheme called GIF for NDN-based WSNs. In this proposal, they added support for push-based WSN traffic. They also extend the scheme with efficient forwarding techniques including flooding scope control, broadcast storm avoidance, packet suppression in order to balance the energy consumption across the network. As shown in [14], data packet flooding is a hot research topic due to the broadcast nature of the CCN-based wireless sensor networks. For this reason, to mitigate this problem, the authors proposed a novel protocol PDLP, named Packet Diffusion-Limited Protocol for CCN-based WSNs for smart cities. The proposed protocol limits flooding of Data packets in a certain range over the networks, and speeds up content download time using a shortest path from a provider to a consumer.

To the best of our knowledge, despite the high potential of CCN, only a few papers have recently addressed the use of this paradigm in wireless multi-hop environments. To this end, in order to support a new forwarding strategy and reduce the amount of energy consumption, we focus in this thesis on how to achieve energy efficiency by proposing a new forwarding strategy in CCN enabled WSNs and an efficient caching strategy. In the next section, we list the existing caching strategy in WSNs.

3.6 In-network caching

In-network content storage has become an inherent capability of content-centric networking architecture which raises new challenges in the use and the provision of the caching placement. Therefore, a trade-off between the network performance and the provisioning cost has to be found [87]. In contrary, this concept has been already used before in CDNs ‘Content Delivery Networks’ [88] and Web Cache [89] which have been widely adopted on the Internet in order to improve the content delivery efficiency and the application-specific overlays solutions. CDNs use local content caching as a key function. However, as an overlay content caching service, CDN may cause significant conflicts with the network traffic engineering since the objectives of content caching and traffic engineering can be different [90].

As one of the most promising potential architecture for the future Internet, Content-Centric Networking has attracted a lot of attention. Comparing to CDNs, CCN integrates the content routing and caching process in the network layer, and can potentially achieve much better performance in the aspects of resource utilization and content dissemination efficiency [90]. Hence, it offers transparent and ubiquitous in-network caches which are the fundamental building block that guarantees efficient content retrieval [11]. Besides, caching nodes in CCN are application independent. By against, the traditional caching system is application dependent and caches use proprietary protocols [58]. In addition to that, caching in CCN can deal with several types of traffic and any node in the network can handle caching. However, old caching technologies are defined for a specific traffic and usually located in a predetermined location.

Among several existing works in the literature, it was demonstrated in [91] that ICN caching mechanism including CCN outperforms traditional caching technologies in terms of network quality of service especially delivery latency, network resource consumption, and end-user experience. Nevertheless, in-network caching poses too many decision challenges related to "content placement" (WHERE to cache the content), ‘content replacement’ (WHICH content is to evict from the cache), and ‘request routing’ (HOW to redirect interests to an optimal cache). In our thesis, we address the caching mechanism in CCN for wireless sensor networks and we propose a novel caching strategy for CCN-WSNs based on a new content placement and replacement policies. But before, we start by studying the in-network caching in CCN for WSNs.

Wireless sensor networks present some limited capabilities. Since they are too small, memory is limited. Further constraints like energy efficiency are also present. Furthermore, since sensor networks are

used to measure real-time data, this data has to be fresh. Then, sensor nodes have to send fresh data and ensure a certain level of reliability. Content caching is a powerful scheme to improve the content delivery performance of wireless sensor networks. Several researchers have been devoted to the cache management problem in the caching nodes of sensor networks. The structure of the CCN mechanism enables the storage of data in every sensor in wireless sensor networks. However, caching the content on all over the nodes in the network is not a good strategy in terms of resource utilization.

Several works have been proposed in WSNs by researchers in order to realize the caching of the data either in some intermediate nodes or at a location nearer to the sink. Indeed providing solutions to optimally caching the data has been a big area to be focused on. In [75], the authors presented a GCCS scheme called Global Cluster Cooperation for wireless sensor networks for caching in wireless sensor networks. GCCS scheme uses a cache discovery algorithm to find the node who has cached the queried data item. It also employs cache admission control, cache consistency, cache replacement mechanism to improve the overall performance of the scheme. GCCS scheme exploits cooperation among sensor nodes and the decision regarding a data item depends upon the value of distance calculated beforehand. They exploited also the grid approach in order to optimize the utilization of energy.

In order to handle and overcome all the constraints set by wireless sensor networks, some researchers addressed the in-network caching in wireless sensor networks.

Gayathri *et al.*, [76] proposed an information-centric scheme for wireless sensor networks using cognitive in-network devices. Then, the routing decisions became dynamic and based on specific Knowledge and Reasoning-observations in the WSNs. Knowledge representation using the <attribute, value> pair, and reasoning using AHP an Analytic Hierarchy Process. These techniques are used at the cognitive device in order to decide on the best data route. AHP can be applied on Quality of Information (QoI) attributes in next-generation WBANs and can provide reliability, better delay, and network throughput over the communication paths.

As discussed in [78], more researchers start to exploit information-centric networking in WSNs since they present the major technique in the sensing layer of the Internet of Things. Authors proposed a collaborative caching strategy for the information-centric wireless sensor network. The proposed strategy consists of three parts: the node betweenness based cache size adjustment, the data replacement frequency

based cache decision, and the content value based cache replacement algorithm. They affirmed that nodes can find a balance between caching performance and storage consumption.

To summarize, many works investigated the problem of caching in CCN. However, some works treated this in CCN-WSNs. For this reason, we will focus on treating the problem of caching in WSNs by addressing CCN features and finding the impact of new parameters.

3.7 Application scenario: traffic on Paul Sabatier University

In this section, we present some of the essential use cases in neOCampus project [20] in which our thesis is involved. These scenarios raise the need for implementing CCN on sensor nodes and proposing new caching strategies that handle content dissemination in order to minimize energy consumption. NeOCampus goal is to make the campus of the university smart by interconnecting all the buildings while saving energy and respecting the environment.

WSNs have been applied in monitoring systems in various fields such as precise agriculture, remote health-care, and animal behavior detection. In this thesis, we consider the example of Paul Sabatier university with many buildings implementing sensor networks to build our use cases. We assume that each building, restaurant, parking and university library is equipped with sensors that measure different pieces of information.

As shown in Fig. 3.7, several cases can be gathered in a smart campus. For instance, sensors at classrooms measure the temperature, the humidity, luminosity, presence, professors availability, energy equipment consumption and contain information about the schedule. Students and staff use their devices to ask for contents and they are interested in getting the requested content without having an idea about its location. Let us consider a sensor network operating in the building $U4$ of the university, in the restaurant and in the university parking. Sensor nodes are widespread on the campus to ensure the communication between different sensors. A User (e.g., student) on the campus is interested in the temperature in classroom 204 in building $U4$. He broadcasts an interest in the network and the node with the corresponding content replies with it. Then, whenever and wherever, a student or a personal staff wants to have an information about classrooms, he can send a query. The sensors network in the restaurant affords information about the state of the restaurant queue and the one operating in parking gives information about available places. The traffic is not the same the whole day. In the rush hours, for example, around 11.30 *am*, a lot of students are

interested in the state of the restaurant queue. Between 7.30 am and 8 am, students arrive at the campus and try to find a parking. Other users look to have an idea about the classrooms temperature. Users who ask for this information later get the response faster since the information was already broadcasted in the network and is available in intermediate nodes.

The future digital campus will be both accessible, sustainable, and smart. It will be full of sensors, autonomous but capable of evolving. The challenge will be in the manner the data is managed. For these reasons, we thought about proposing a new caching strategy to manage efficiently this amount of network data on the campus.

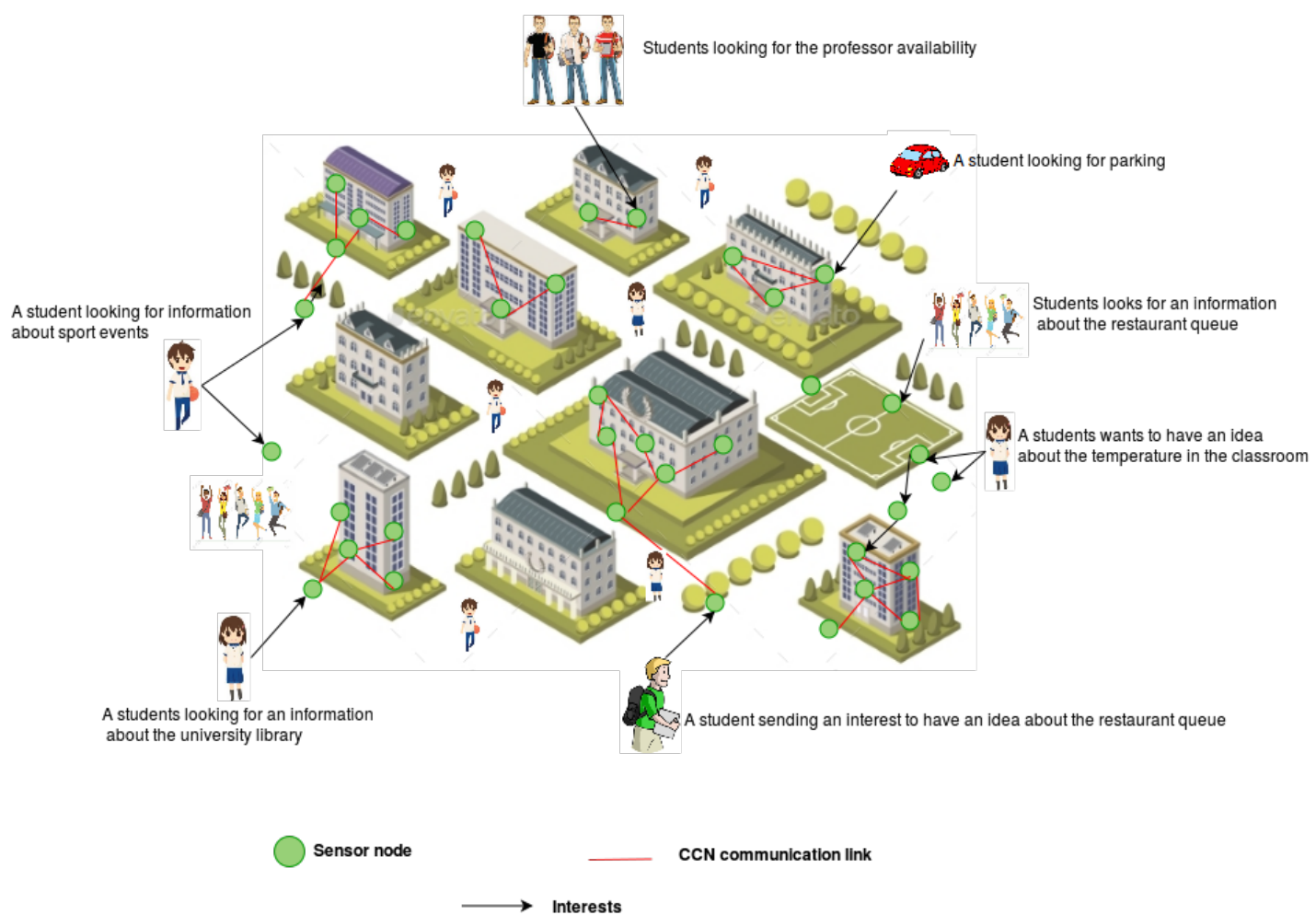


Figure 3.7. Wireless sensor network deployment and CCN communications in a campus.

3.8 Conclusion

The data transmission is done from the sensor nodes inside each deployment area with a very big redundancy level which leads to energy waste. To avoid this, data-centric routing protocols were designed. Naturally, WSNs are regarded as data-centric networks. Hence, the activation of content-centric networking paradigms seems to be a good idea. However, enabling CCN in WSNs presents a lot of challenges especially in terms of memory and energy efficiency. This concept has been the subject of several researchers in the last few years. Some studies have shown that CCN paradigm could be a good candidate for wireless sensor networks and it was demonstrated that implementing a lightweight adapted version of CCN in WSNs may resolve a lot of issues. In our case, this integration will help in managing the data exchange in the campus. In this chapter, we cited some related work and we motivated our choice of CCN to better highlight the relevance of its exploitation and its suitability in a sensor network. We listed its advantages and its limitations in WSNs and we detailed the fields in which we are going to propose some enhancement. Finally, we presented a potential application scenario for the work carried out in this thesis.

CHAPITRE 4

Data Freshness in Content-Centric Wireless Sensor Networks

Contents

| | | |
|------------|--|-----------|
| 4.1 | Introduction | 46 |
| 4.2 | State of the art | 46 |
| 4.2.1 | Data freshness and reliable data delivery in WSNs | 46 |
| 4.2.2 | Reactive and proactive routing in Wireless Sensor Networks | 48 |
| 4.3 | Content lifetime in CCN for WSNs | 49 |
| 4.3.1 | Principle of DFCCN-WSNs | 50 |
| 4.3.2 | Motivating example | 50 |
| 4.4 | Content-update strategies | 52 |
| 4.4.1 | Proactive approach | 53 |
| 4.4.2 | Reactive approach | 54 |
| 4.5 | Performance evaluation | 57 |
| 4.5.1 | Simulation set-up | 57 |
| 4.5.2 | Evaluation metrics | 57 |
| 4.5.3 | Simulation results | 58 |
| 4.6 | Conclusion | 63 |

4.1 Introduction

Sensor networks are used in several applications, including habitat and ecosystem monitoring, seismic monitoring, rapid emergency response, perimeter security and surveillance, groundwater contamination monitoring, and etc. Wireless sensor networks are used to achieve either continuous monitoring or event detection in the area of interest. In continuous monitoring applications, nodes transmit periodically their measured data to the sink. These applications require fresh data at the sink. Therefore, data freshness is important in such scenarios.

Moreover, as discussed in the previous chapter, sensor nodes have limited power, storage, and processing capability, which requires the need for a lightweight protocol such as CCN. However, CCN default caching approach lacks a data freshness mechanism, while IoT data are transient and frequently updated by the producer which imposes stringent requirements in terms of information freshness. In this chapter, we focus on the lifetime of the content exchanged in the network and we explain the idea of its integration in CCN for WSNs to better highlight the relevance of its exploitation. In order to respond to the needs of continuous-monitoring applications and improve reliability in CCN enabled WSNs, we implement DFCCN-WSNs 'Data Freshness aware Content-Centric Networking in Wireless Sensor Networks' a protocol that integrates the content lifetime. We also propose proactive and reactive strategies for content update in CCN for sensor networks.

4.2 State of the art

4.2.1 Data freshness and reliable data delivery in WSNs

Data freshness concept has a relation with how old is the data? Is it fresh enough with respect to user expectations? Data freshness guarantees that the old messages cannot be relayed by any node. There are various definitions of data freshness in the literature, depending on the applications where they are used, as well as different metrics to measure them. In [92], they proposed some metrics to measure data freshness such the currency metric, which measures the time elapsed since the source data changed, the obsolescence metric, which measures the number of updates since the data-extraction time, and the freshness-rate metric, which measures the percentage of extracted elements that are up to date.

Data freshness has been identified as one of data quality attributes in WSNs. Continuous monitoring

applications require periodically refreshed data at the sink node. So it is important for the data to reach the sink node within a certain threshold. WSNs are generally organized in a multihop topology since messages travel by multiple hops, it is important to have high reliability.

For instance, as presented in the neOCampus application scenarios [20], students on the campus require fresh data concerning the status of the queue in front of the restaurant or the availability of parking places. When they send a query asking for this at 11 am, they need a data measured in at 11 am or a data considered still fresh, not a data measured at 9 am for example. This depends on the information lifetime. However, resource constraints present a major challenge in achieving data freshness along with reliable data delivery of packets.

So far, data freshness was measured only in terms of latency or delay of packets received at the base station [93]. To improve data freshness, the packets should reach the sink reliably and the latency should be minimized. Therefore, when a student on the campus asks for the temperature of the classroom if the content arrives to the student within a specified time interval, it stays fresh in intermediate nodes for a longer time. Some applications consider data received after a certain threshold as stale and do not use it for evaluation or analysis. So, the consideration of freshness is also application-dependent. Moreover, in sensor networks, data lifetime differs from data to another and from application to another. For instance, in [92], they proposed a taxonomy based on the nature of the data and the type of application to define data freshness. To ensure data reliability then data freshness, too many mechanisms such as data-aggregation exists in WSNs [94]. At the same time, data freshness ensures reliability and low latency since users may recover fresh data from the sink. The position of the sink also has a significant impact on data freshness in wireless sensor networks [95]. If the sink is in the center the data is yield fresher.

In addition to that, in WSNs, data freshness is involved in the concept of network security. In fact, to fulfill the security requirements in WSNs, data integrity, source authentication, data confidentiality, and data freshness are taken into consideration by applying cryptographic algorithms [96]. Data integrity guarantees that the message has not been altered during the propagation. However, if data aggregation is employed, alternations could happen since it is not possible to have end-to-end data integrity. Consequently, data freshness protects data aggregation from reply attacks [96].

Policies for maintaining data freshness are traditionally either push-based or pull-based. Push-based

policies involve pushing data updates without no request; they may not scale to a large number of users. Pull-based policies require users to send requests to check for updates; their effectiveness is limited by the difficulty of predicting when to ask for updates.

Guaranteeing the data freshness avoids also the re-routing of the content in order to update stale information. Consequently, the sink or the user depending on the architecture type do flood the network with interests in order to get fresh content which economizes energy and avoid congestion and overhead [97].

4.2.2 Reactive and proactive routing in Wireless Sensor Networks

Many routing, power management, and data dissemination protocols have been specifically designed for WSNs where energy awareness is an essential design issue. Routing protocols in WSNs can be also classified into three categories, proactive, reactive, and hybrid, depending on how the source finds a route to the destination. In proactive protocols, all routes are computed before they are really needed, while in reactive protocols, routes are computed on demand [98]. In other words, proactive protocols periodically monitor peer connectivity to ensure the availability of any path amongst active nodes. On the other hand, reactive protocols establish paths only in response to a query, or an event; meanwhile, sensors remain in the idle state in terms of routing behavior. And the hybrid category combines both proactive and reactive. Indeed, proactive and reactive protocols are associated with different costs, in terms of resource overhead (e.g. energy or bandwidth) and non-functional guarantees (e.g. end-to-end delay, or time to repair) [99].

When nodes are static, it is preferable to use proactive protocols since in reactive protocols an important amount of energy is used when establishing route discovery and setup [98]. Reactive protocols were chosen as the best candidate for mobile ad hoc networks due to node mobility. In contrast, proactive routing is considered more attractive in sensor systems [100]. In [101], they proposed an Energy-Aware Routing protocol, a destination-initiated reactive protocol that aims to increase the network lifetime. Direct Diffusion [47] and TEEN [102] are also proposed as reactive routing protocols, while SPIN [45] is a proactive routing protocol. Direct diffusion and SPIN are data-centric protocols. For instance, in SPIN, sensors proactively advertise their measurements, via a 3-stage hand-shake protocol, to disseminate them across the network [100]. However, Direct Diffusion is a reactive protocol where a data query, or interest, determines the flow of data from one or more source nodes to a sink [47].

Proactive routing protocol for large-scale networks has proven a worthy adversary of its reactive counterparts for wireless sensor networks since proactive protocols disseminate state across the network, in contrast to reactive protocols who reveal state only upon request [100].

Finally, the designer of the application in function of its requirements has the choice to choose between the two routing models. In our thesis, we implement both strategies for content update and we study their impact on the delay and energy.

4.3 Content lifetime in CCN for WSNs

In [103], the concept of data lifetime was introduced for the first time in content-centric networking when the IETF community was describing the header format for CCN packets. In this draft, they propose to integrate lifetime for interests and contents. Content lifetime indicates the validity of a content expressed in seconds. After that time, the data will be considered as not valid and discarded or ignored if stored in the local cache.

In the RFC 7927 [104], the authors discussed the problem of staleness cached copies in the content store. They argue that ICN paradigms such as CCN should provide a staleness verification algorithm to ensure the freshness of cached content. Besides, they agree that a direct and indirect approach should be considered to ensure this and a timestamp that indicates the data validity should be added to the content. A direct approach enables to look directly to the timestamp and deduce the data staleness. On the other hand, for the indirect approach, each cache checks the publisher of the content to decide to cache it or discard it. In the majority of machine-to-machine, Internet of Things and wireless sensor network scenarios, a fresh version is asked depending on the application requirements.

Recently, in IETF internet draft [105], they proposed CCNinfo that discovers information about the network topology and in-network cache in Content-Centric Networks. In addition to that, CCNinfo premises also new information about the cache such as lifetime and expiration time per cache or chunk.

For content-centric wireless sensor networks, every node in the network has its own content store in which it caches contents. Initially, nodes create their content measured locally (by adding the content measured or provided by the node locally). When a user sends an interest it starts by checking its CS and hereafter it transmits the interest to the other nodes in the network and the node with the corresponding content responds with the data located in its cache. Absolutely, if the data lifetime is not checked, there

is a risk that when it reaches the user it may be already expired and does not satisfy the user expectation. Since we are working with wireless sensor networks and since our applications are neOCampus use cases oriented, our goal is to provide campus users with fresh data.

In this context, we propose DFCCN-WSNs, a '*Data Freshness aware Content-Centric Networking for Wireless Sensor Networks*' in which we integrate the lifetime of the content recommended in the last version of CCN [103] but this time, it is for the adapted CCN for wireless sensor networks. We study its impact on the global system in terms of end-to-end delay and energy consumption.

4.3.1 Principle of DFCCN-WSNs

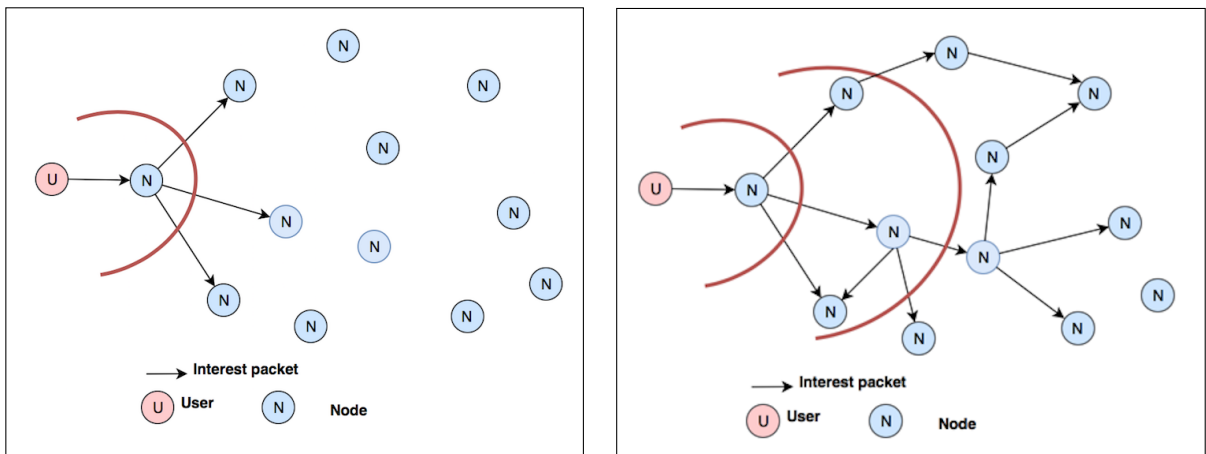
In DFCCN-WSNs [19], a user broadcasts an interest packet in the network. When the request arrives at a node, it begins by checking if it has the adequate content in its CS, if yes it checks if its lifetime did not expire yet. If the content is fresh, it responds with the content. Then, in the proposed protocol, the content freshness is taken into consideration. No obsolete data is routed in the network.

Therefore, taking into consideration data freshness when matching an interest with the corresponding content is important in order to ensure a certain level of QoS in the network. Certainly, if the node finds the requested content and if it is still fresh, the intermediate node can send it. Otherwise, the user has to address his interest in the source to recover the data.

4.3.2 Motivating example

For the sake of illustration, we consider a motivational example in Figure. 4.1, where different nodes and their range in the network are presented. In this example, U represents the user who broadcasts interests in the network. Nodes N measure the temperature, the humidity, the luminosity, the presence, the queue length in front of the restaurant and states of shutters in the classroom. Figure. 4.1 shows how the interest is broadcasted in the network.

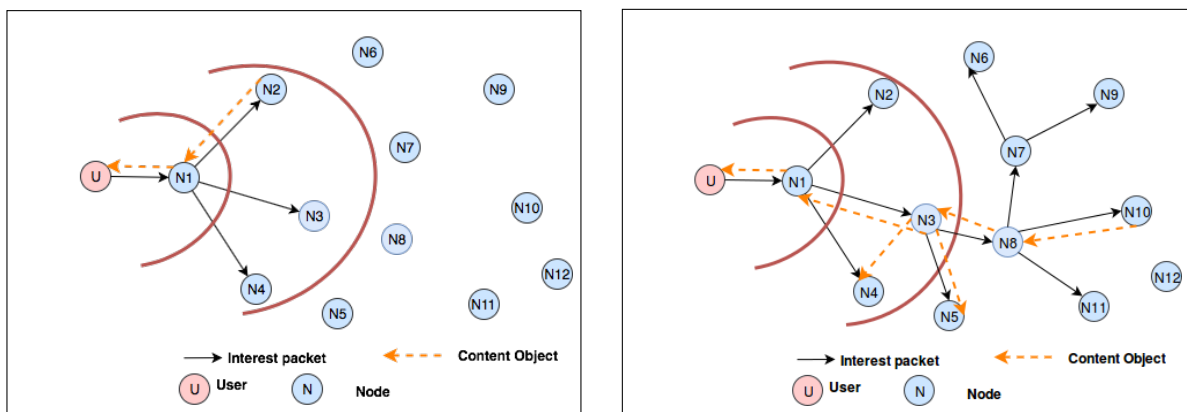
The user in this case wants to have an information about the temperature of the classroom 204 in the building $U4$. To request this content, as described in Fig. 4.2, the user sends an interest and waits to recover the content. Since the nodes $N1$ is in the range of the user, it checks if it has the information. If it does, it answers. Otherwise, it broadcasts the interest. In this example, as shown exactly in Fig. 4.2(a), node $N2$ has the content corresponding to the temperature in its cache (measured locally). Then, it responds.



(a) The user sends its interest in the network.

(b) The sensor node receives the interest and broadcasts it to its neighbor after checking its CS.

Figure 4.1. Interests broadcast in CCN-WSN network.



(a) The user broadcasts an interest for the temperature and node N2 has the corresponding content.

(b) The user broadcasts an interest for the humidity and node N10 has the corresponding content.

Figure 4.2. Expressing interests for temperature and humidity.

Once the content takes the path to the user, a copy of each will be cached in the nodes content store. When the user U asks for the same content, he recovers the contents from an intermediate node, if they are still available (if they were not deleted because of cache size and content replacement) and the interest is not broadcasted further. In unmodified CCN [70], it responds with the information directly without any

verification and the information could be stale.

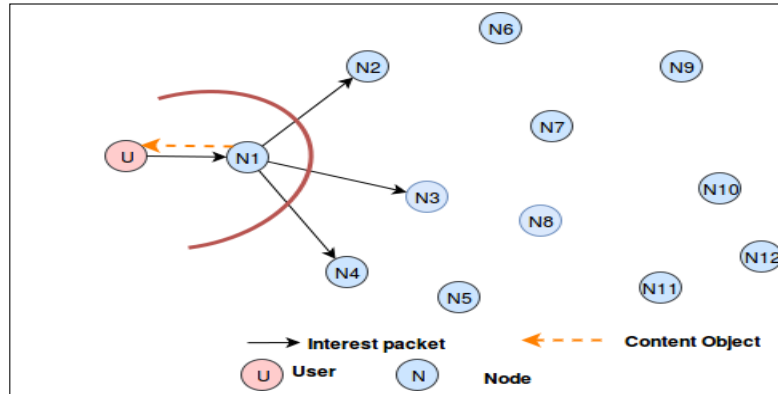


Figure 4.3. The user recovers the content from intermediate nodes.

In DFCCN-WSNs, the node checks if the content is still fresh (its lifetime did not expire yet), if yes it responds with the content, if no it checks if the content was created locally this means that the data is expired (if the node does not update its content once expired which is not the case) and if it was relayed by a node it broadcasts the interest. For example, the lifetime of the temperature can correspond, for instance, to 30 minutes. The temperature will not change drastically within 30 minutes. In this example, the temperature was measured by node $N2$, so if the data is still fresh, node $N2$ broadcasts the content in the network and satisfies the user interest. If the user sends an interest for the humidity which measured by $N10$ as depicted in Fig. 4.2(b), the same process is repeated and the content is recovered from node $N10$. As illustrated in Fig. 4.3, when the user asks for humidity, he receives the data from node $N1$. Otherwise, the node broadcasts the interest in the network and the user gets the data from the source ($N10$) if the content was updated. So by introducing the concept of the lifetime, all the data exchanged on the network remain fresh and a certain level of QoS is achieved.

4.4 Content-update strategies

In DFCCN-WSNs, a user broadcasts an interest packet in the network. When the request arrives at a node, it begins by checking if it has the adequate content in its CS, if yes, it checks if its lifetime did not expire yet. So, data freshness has to be taken into account when matching an interest with the corresponding content. If the node finds the right content and if it is still fresh, the intermediate node can send it. Otherwise, the

user has to address his interest in the source which creates the content to recover the data. A two-way ranging technique has gained popularity in WSNs: the first one is doing the update of all contents even in intermediate nodes (proactive approach) or waits for an interest to update contents (reactive approach).

4.4.1 Proactive approach

In the proactive approach, all contents created by the nodes are updated when they expire regardless of the traffic (reception of an interest). So, when a popular content expires in the source node, the update is realized automatically. Intermediate nodes broadcast an interest for updated contents when a popular content in its CS expires. Therefore, we enhance DFCCN-WSNs by adding this proactive content-update mechanism. The resulting protocol is called PDFCCN-WSNs 'Proactive Data Freshness aware Content-Centric Networking for Wireless Sensor Networks'.

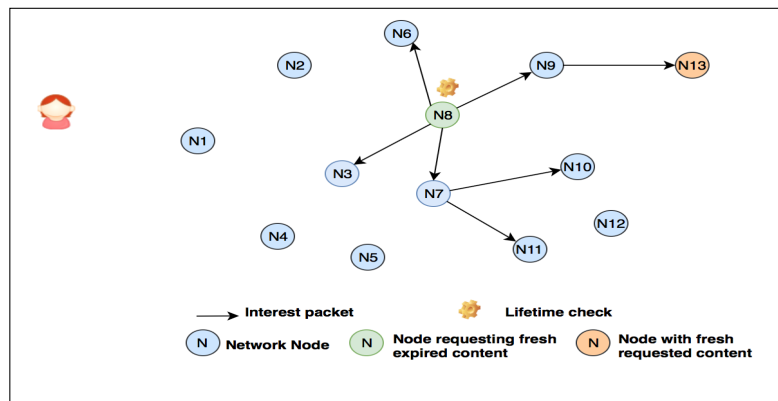


Figure 4.4. When the node $N8$ realizes that the data lifetime is expired, it broadcasts an interest.

Let consider that node $N13$ has the content describing the queue length in front of the university restaurant, as shown in Fig. 4.4. In the proactive strategy, once the node $N8$ discovers that the corresponding content lifetime expires, it broadcasts an interest on the network asking for this content. $N13$ answers with the corresponding content as depicted in Fig. 4.5 .

The advantage of the proactive approach is that the content will be immediately available so the delay is optimized when replying with a content. This also helps in optimizing flooding ie. nodes find contents available when receiving interests from users. For instance, it was demonstrated in [106] that proactive approaches achieve significant positive trends for the delivery ratio and power consumption when using a

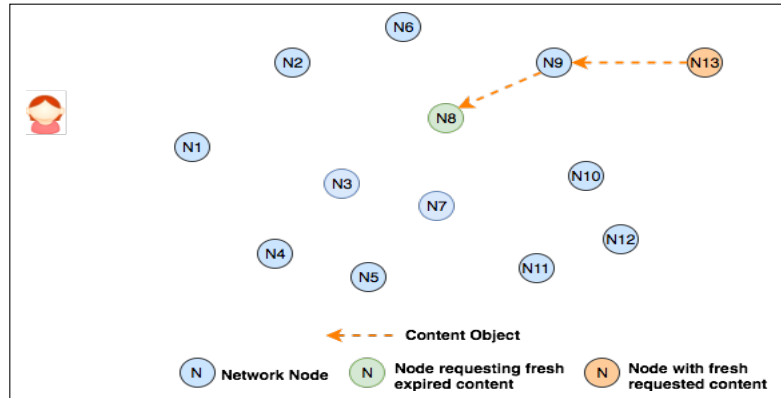


Figure 4.5. When the node $N13$ receives an interest for the queue length, it responds with this content.

proactive approach. When the content lifetime is large, the proactive approach may realize energy efficiency. In other words, the proactive content-update approach may anticipate and adapts for the high traffic just before it actually occurs, avoiding adaptation in the period in which coordinating a mode change is more challenging and many packets can be lost.

4.4.2 Reactive approach

For the reactive approach, in contrast, nodes send interests for a fresh content depending on the traffic (reception of an interest) as shown in Figure. 4.6. Hence, the content update is realized once an interest is received from a user. So when a node receives an interest, if the data is still fresh it responds the user with this content. Otherwise, it verifies if the content was created locally or received from another node. If it was created locally, content is not going to be sent since it is expired (if the node did not refresh its measure because of a problem in the node). Otherwise, it removes the content from its CS and broadcasts it in the network to look for it in the nodes in which it was created if it is updated. So we extend DFCCN-WSNs by RDFCCN-WSNs '*Reactive Data Freshness aware Content-Centric Networking for Wireless Sensor Networks*'.

For instance, once the user wants to have an information about the queue state in front of the restaurant. It broadcasts an interest in the network. Considering that this information was asked before, it means that it was cached in intermediate nodes. As shown in Fig. 4.6, the interest is broadcasted and since intermediate nodes cached a copy before, node $N1$ checks the content lifetime. If it is still fresh, it responds with it. Otherwise, the interest is broadcasted in the network and the source node or an intermediate node

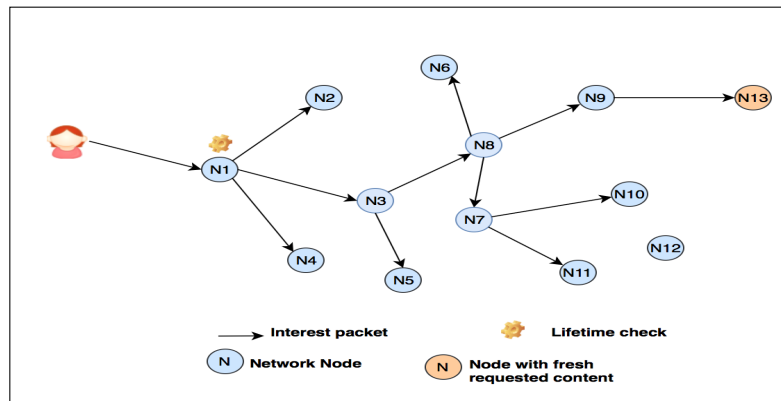


Figure 4.6. The user broadcasts an interest to have an idea about the queue state in front of the restaurant.

detaining a fresh content answers with this latter. All intermediate nodes who cached beforehand the asked content establish a lifetime check. In the scenario, depicted in Fig. 4.7, the content is recovered from the source node $N13$.

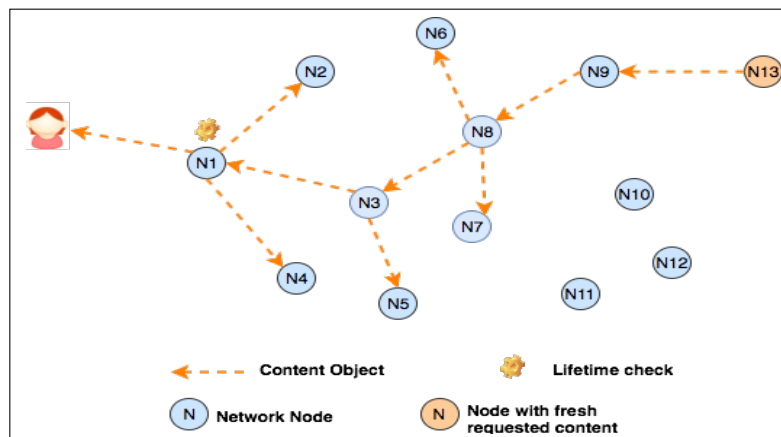


Figure 4.7. The content is recovered from the source node $N13$.

When the traffic is light and when the lifetime of contents is large, it would be intuitive that reactive approach would be energy efficient and inefficient when the traffic is heavy (which is the case in the scenario described previously). In addition to that, it was shown in [106] that the reactive approach tries to compensate for the low delivery ratio by increasing the transmission power, which increases power consumption.

Algorithm 1: Requesting a fresh content object

```

input   :  $I$ : Interest message
            $CS$ : Content Store
            $PIT$ : Pending Interest Table
            $FIB$ : Forwarding Interest Table
            $data\_lifetime$ : Content lifetime

output  :  $CO$ : Content Object

1: Receive an interest message  $I$ ;
   //If matching content found in  $CS$ , checks its lifetime, if it did not expire yet, send
   the corresponding content
2: if ( $Content \in CS$ ) then
3:   if ( $data\_lifetime$  did not expire yet ) then
4:     Content Object is transmitted;
5:   else
6:     //Check if it was created locally or received
7:     if ( $locally$ ) then
8:       Find out that the content is expired;
9:     else
10:      Remove Content Object from  $CS$ ;
11:      Forward interest;
12:      if ( $updated$ ) then
13:        Transmit;
14:   else
15:     if ( $matching\ Content \in PIT$ ) then
16:       Add the interest to  $PIT$ ;
17:     else
18:       if ( $matching\ Content \in FIB$ ) then
19:         Create an entry in  $PIT$ ;
20:          $I$  is sent to the destination registered in  $FIB$ ;

```

The principle of the two approaches is described in Algorithm 1. The difference is in the manner the update is done; depending on traffic (reactive) or without traffic (proactive). Indeed, for the reactive approach, the user broadcast an interest and the update is done if the content became stale. Otherwise, when a node realizes that a popular content expired, it looks for it to update it.

4.5 Performance evaluation

In this section, we examine the performance of RDFCCN-WSNs and PDFCCN-WSNs to see the impact of the content update on the network performances. The novelty in our scheme is the support of multi-user scenarios since previous works consider only one user via one sink [70]. For this, we set different point of entrance in the network and behind this points, a lot of users (students in our use case) broadcast their interests. Hence, the performance of our solution is evaluated under a different number of users. We set 5 entrance points in the network. Behind entrance point 1, there exist 30 users, behind entrance point 2, 90 users, behind entrance point 3, 30 users, behind entrance point 4, 30 users, and finally behind entrance point 5, 10 users. Entrance points serve as point receiving the interest and pulling them into the network. These points realize aggregation of interests when they have similarities.

4.5.1 Simulation set-up

For the implementation of our solution, we chose to start with the code of CCNx_Contiki [70] since it contains a framework that implements CCN in WSNs and we modify it to follow the requirements of our approaches. Contiki [107] is an open source operating system for memory-constrained embedded systems and wireless sensor networks. It is highly portable and ported to more than twelve different microprocessor and microcontroller architectures. Contiki is designed for microcontrollers with a very limited memory size.

We consider a wireless sensor network deployed on a surface of $300\text{ m} \times 300\text{ m}$. The number of nodes varies from 30 to 120. All the sensor nodes are static and have the same radius detection. In addition, all communication links are bidirectional. We consider that we have three different types of content generated by nodes in the network $N1$, $N2$, and $N3$. Contents have a limited lifetime. When the lifetime expires, the nodes update their content. Lifetimes are 80s , 60s and 30s corresponding to $N1$, $N2$, and $N3$ respectively. We suppose that the clock nodes are synchronized [108]. Simulation parameters are detailed in Table 4.1.

4.5.2 Evaluation metrics

For the evaluation of the proposed approaches, we chose the following metrics:

- Propagation delay: Time between the broadcast of the interest and the recovery of the corresponding content object.
- Energy consumption: energy consumed by all nodes in the network.

| Simulation Parameters | Value |
|-----------------------|-----------------|
| Number of nodes | 30, 60, 80, 120 |
| $N1$ lifetime | 80 <i>s</i> |
| $N2$ lifetime | 60 <i>s</i> |
| $N3$ lifetime | 30 <i>s</i> |
| Poll interest point 1 | 15 <i>s</i> |
| Poll interest point 2 | 10 <i>s</i> |
| Poll interest point 3 | 20 <i>s</i> |
| Poll interest point 4 | 10 <i>s</i> |
| Poll interest point 5 | 55 <i>s</i> |

Table 4.1. Simulation parameters

4.5.3 Simulation results

We start by exploring the results of when there is only one user in the network and then we present the results for multi-users.

4.5.3.1 One user

In this scenario, we have one user who broadcasts its interests through one entrance point to the network and 30 nodes generating ($10 \times N1$, $10 \times N2$, and $10 \times N3$).

| Delay | 1 st round | 2 nd round |
|-------|-----------------------|-----------------------|
| N1 | 165 <i>ms</i> | 156 <i>ms</i> |
| N2 | 500 <i>ms</i> | 447 <i>ms</i> |
| N3 | 274 <i>ms</i> | 313 <i>ms</i> |

Table 4.2. Delay when applying the RDFCCN-WSNs update strategy

An interesting observation comes from Table 4.2 that shows that for the content $N2$, the delay is equal to 500 *ms* and this is due to the network deployment. So, this can be explained by the fact that nodes that contain the content $N2$ are not in the range of the user. As for nodes that contain the content $N3$, they are in the range and very close to the user.

Ordinarily, during the 2^{nd} round, contents will be available in the content store of the nodes that are close to the user so the delay must be minimized compared to the 1^{st} round. Hence, the user may get the requested content from many nodes (the content's source or the intermediate nodes). However, by taking into account the content lifetime, this may be different. It will depend on the approach that we are going to choose.

The user sends an interest for the content $N2$ at $t=75$ s and the lifetime of the content $N2$ is 60 s. When RDFCCN-WSNs is used, it is found in the content store of intermediate nodes with expired lifetime so the interest is relayed to the source nodes which explains the delay for the content $N2$ during the 2^{nd} round. For the content $N3$ also, its lifetime is about 30 s and the interest was broadcasted at $t=90$ s. For $N1$, the delay is reduced because the content is found in intermediate nodes but the reduction is not so important because the content was already not so far from the user.

Based on the values reported in Table 4.2, we notice that with RDFCCN-WSNs, we lost a little bit in terms of delay but we are sure that the content sent on the network is still fresh.

| Delay | 1^{st} round | 2^{nd} round |
|-------|----------------|----------------|
| N1 | 165 ms | 156 ms |
| N2 | 500 ms | 187 ms |
| N3 | 274 ms | 197 ms |

Table 4.3. Delay when applying the PDFCCN-WSNs update strategy

Given the results shown in Table 4.3, we observe that when we adopt PDFCCN-WSNs (realizing the update without waiting for a user interest), the delay decreases compared to the results presented in Table 4.2. For the content $N2$, we gain 260 ms in terms of delay. Thus, for the content $N3$, a gain of 116 ms is realized.

4.5.3.2 Multiple users

In this part, we investigate the impact of supporting multiple users. Due to simulator constraints, we consider entrance points through which user interests are injected into the network. To this end, in this scenario, we have 180 users behind 5 entrance points and 120 nodes that measure 3 contents ($N1$, $N2$, and $N3$) under the same simulation parameters. Note that the old CCNx_Contiki [70] did not support multiple users.

Once data collection starts, for every interest sent by a user, only source nodes have a matching content, may satisfy the user expectation. As long as a content object is sent over the network, it is kept in the CS of the intermediate nodes. From the second round, the content is available also in intermediate nodes so the user interest is satisfied under better and bounded delays.

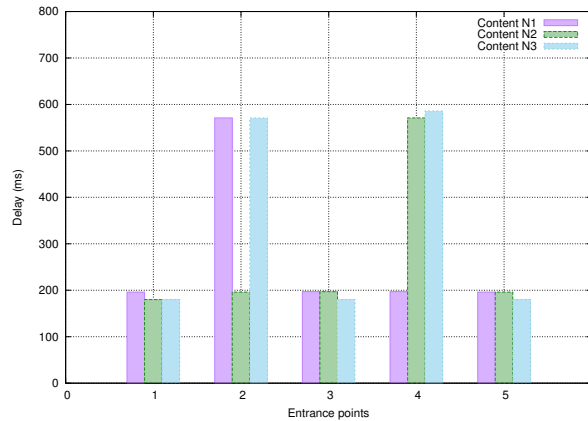


Figure 4.8. Delay for different users during the 1st round for different types of generated content.

Figure. 4.8 shows different delays for different users in the network. We notice that for users behind point 2 and 4, the delays are very important since they were the two first users that broadcast their interest on the network at $t = 10$ s. At that time, only source nodes had their corresponding content. The user behind point 2 gets quickly the content $N2$ because a node that contains this content is in its range. The same applies to the content $N1$ for user behind entrance point 4. Once contents are exchanged over the network, the users who broadcast their interests later can get a response faster. For instance, the user behind entrance point 1 waited only for 196 ms to get the content $N1$. So, if we have multiple users in the network, the delay may decrease since other users broadcast frequently interests.

Usually, during the 2nd round, almost all the nodes have the three different contents in their content store. As a consequence, the delay is not important. However, as depicted in Fig. 4.9, user behind entrance point 1 waits for 707 ms to get a response for the content $N3$. This is because of user 1 (user behind entrance point 1) broadcast an interest for the content $N3$ at $t = 45$ s, while content $N3$ lifetime is about 30 s. Hence, the nodes that have the content $N3$ in their CS have outdated content. Since the RDFCCN-WSNs approach is adopted here, user 1 waits for more time to get a response from source nodes. Consequently, the delay

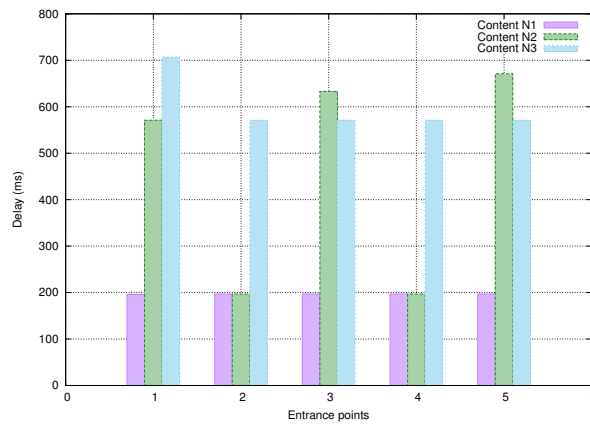


Figure 4.9. Delay for different users during the 2nd round with the reactive approach RDFCCN-WSNs.

increases.

For the content $N1$, the delay is smaller than other contents since its lifetime is significant and it was requested before its expiration.

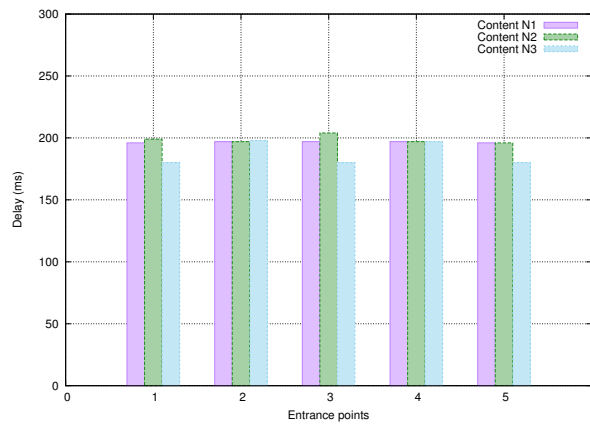


Figure 4.10. Delay for different users during the 2nd Round with the proactive approach PDFCCN-WSNs.

When PDFCCN-WSNs is adopted, users may get a response faster. Figure. 4.10 shows the delay for different users during the second round if content update is realized in intermediate nodes. We notice that the delay for the content $N3$ for user 1 became 180 ms . So PDFCCN-WSNs approach realizes a gain of 527 ms . As far as for the content $N2$ for user 3 (behind entrance point 3), with RDFCCN-WSNs the delay is about 633 ms and after applying PDFCCN-WSNs it becomes 204 ms .

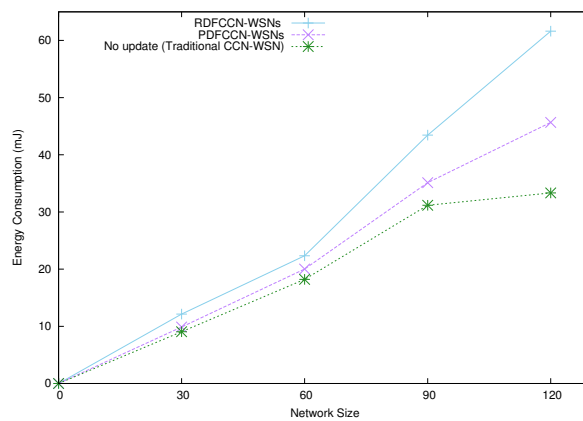


Figure 4.11. Energy Consumed by the network for different approaches for 2 rounds.

From the curves shown in Figure. 4.11, we noticed that the average energy consumption for the three methods increases with the density of the network, which is expected since every node consumes energy. The RDFCCN-WSNs approach (concerning update when we receive an interest) consumes more energy because when content expires in an intermediate node, it has to look for content in source node by broadcasting the interest which consumes energy. However, traditional CCNx_Contiki does not consider content lifetime so content will be sent by the intermediate node without considering its freshness. So given the results shown in Fig. 4.11, we observe that we lost a little bit in terms of energy, for instance, 3 mJ for 30 nodes. But, a gain in term of reliability is realized since content freshness may satisfy the user expectation. Furthermore as mentioned before, in WSNs, continuous monitoring applications require refresh data at the sink node. So, it is important for the data to reach the sink node within a certain threshold. We may notice that for the PDFCCNs-WSNs approach (update even in intermediate node) consumes less energy than RDFCCNs-WSNs. This because when a node receives an interest from a user, if the content object is available on the node, it will be always fresh since the node updated it when it expires so it does not have obsolete content in the node. Hence, a gain in terms of end-to-end delay is achieved. PDFCCNs-WSNs achieves a gain of 20 mJ in terms of energy consumption comparing to RDFCCNs-WSNs. We note also that in the case of multi-users scenarios, the energy consumption increases with the number of interests, which is reasonable since every interest consumes energy.

4.6 Conclusion

Content lifetime is an important content criteria that was not well investigated in the first versions of information-centric networking paradigms. Since we chose to apply CCN in WSNs and since WSN provide monitoring applications requiring fresh data, we could not move forward without dealing with data freshness in CCN for WSNs. Taking into account the data freshness ensures a certain level of network reliability. Besides, it provides new criteria for content eviction from caches, hence it ensures an important cache diversity and cache hit. However, we argue that the need of a fresh or obsolete content depends on the application requirements. Sometimes, some applications aim to retrieve all the contents even the obsolete ones. In the case of the campus, we suppose that students require fresh data all the time. Therefore, we propose two approaches to ensure the availability of fresh content. In this chapter, we started by presenting DFCCN-WSNs which integrated the content lifetime in sensor networks. Besides, we described the two proposed approaches for the content update: the proactive and the reactive strategies. We also demonstrated that PDFCCN-WSNs and RDFCCN-WSNs may realize user interest satisfaction by decreasing the response delay and energy consumption. Finally, we showed that our approaches can support multi-user scenarios, unlike the old CCNx_Contiki. In the next chapter, we will address the forwarding issue in Content-Centric for Wireless Sensor Networks.

CHAPITRE 5

A Duty-cycling Approach for Content-Centric Wireless Sensor Networks

Contents

| | | |
|------------|--|-----------|
| 5.1 | Introduction | 66 |
| 5.2 | Duty-cycling schemes for WSNs | 66 |
| 5.2.1 | Definition of the duty-cycle | 66 |
| 5.2.2 | Taxonomy of duty-cycling schemes | 67 |
| 5.3 | Energy models in CCN | 69 |
| 5.4 | System Model | 70 |
| 5.4.1 | Proposed CCN node model | 70 |
| 5.4.2 | Proposed energy model | 73 |
| 5.4.3 | Numerical analysis of the energy model | 77 |
| 5.5 | Proposed duty-cycle plan | 78 |
| 5.5.1 | Principle | 78 |
| 5.6 | Performance Analysis | 80 |
| 5.6.1 | Simulation set-up | 80 |
| 5.6.2 | Evaluation metrics | 82 |
| 5.6.3 | Simulation results | 82 |
| 5.6.4 | Discussion | 87 |
| 5.7 | Conclusion | 88 |

5.1 Introduction

One of the major issues in WSNs is energy consumption because network nodes are usually battery powered. In ICN, distributed in-network caching can alleviate energy consumptions thanks to a diffused presence of content copies within the WSN. Consequently, building CCN enabled WSNs may also help in optimizing energy efficiency [70]. A lot of schemes were applied in WSNs to save energy [27]. In our thesis, we thought about a duty-cycle scheme by reducing the activity of certain nodes when they are not satisfying a certain amount of interests.

In this chapter, our objective is to investigate the minimum energy consumption that CCN achieves while ensuring a high-interest satisfaction rate (a fraction between the satisfied interests and generated interests in the network). Hence, we come up with ADDC-CCWSN an 'Adaptive and fully Distributed Duty-Cycle for Content-Centric Wireless Sensor Network' mechanism [109]. ADDC-CCWSN aims to reduce the activity of nodes with a high percentage of unsatisfied interests in their PIT. We argue that the approach can be applied (with some modifications) to any ICN architecture that works as a network of caches in a pull mode.

5.2 Duty-cycling schemes for WSNs

Duty-cycling is a mechanism that enables the reduction, the control, and the adaptation of the duty-cycle taking into consideration a defined instant of awake and neighbor synchronization. It has become a fundamental mechanism in the design of wireless sensor networks [110]. Due to its importance, too many approaches have emerged. The objectives behind proposing this type of mechanism in WSNs are to reduce energy consumption and to extend the network lifetime.

5.2.1 Definition of the duty-cycle

A number of definition for duty-cycle exists. They are all functionality purpose oriented. Generally, duty-cycle is presented as the proportion of time during which a component, device, or system is operating. For instance in [111], Vigorito *et al.*, presented the duty-cycle as the percentage of awake time of a node. Furthermore, in [112], the authors define duty-cycle as the ratio between the active period and the full active/dormant period. Whereas, in [113], they measured duty-cycle as the ratio of the listening period

length to the wake-up period length which gives an indicator of how long a node spends in the listening period. Concerning [114], they consider duty-cycle as the period where the node is active.

Finally, we note that several definitions converge since some of them are just relaxations of others and the majority consider that the duty-cycle presents the activity period of a node. Likewise, this is the definition that we consider in our work which we present in Fig. 5.1.

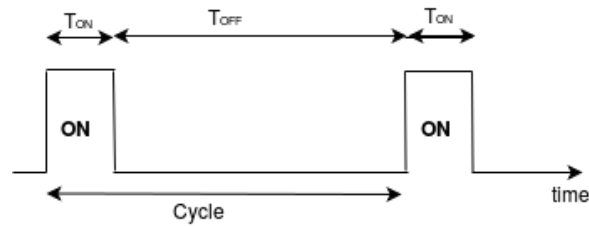


Figure 5.1. An example of a duty-cycle of 20%.

In spite of different existing definitions, a balanced duty-cycle size must be achieved in order to avoid higher latency and higher transient energy due to start-up costs in all techniques [113].

5.2.2 Taxonomy of duty-cycling schemes

As shown in Fig. 5.2 and as discussed in [115], duty-cycling schemes may be classified into synchronous semi-synchronous, and asynchronous depending on the mechanism that coordinates the mote schedule.

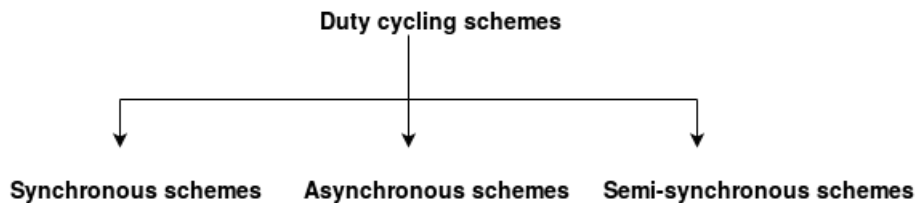


Figure 5.2. Taxonomy of duty-cycling schemes.

5.2.2.1 Synchronous schemes

In synchronous schemes, nodes have the same clocks. They must exchange synchronization information to ensure this. In this category, two schemes are found, the rendezvous and the skewed/staggered. In the

rendezvous, all the nodes turn their radio on and off at the same time. The synchronization is very hard to ensure even between neighbors. Then, the use of synchronized protocols is essential sometimes. For instance, RT-Link [116] is a TDMA based link layer protocol designed to run on sensor nodes and incorporates GPS receiver for clock synchronization. However, this scheme caused data forwarding interruption problem since sometimes nodes may be sleeping when a data arrive. In order to overcome this problem, staggered or skewed schemes were proposed. Indeed, these types of schemes are topology dependent and incur in topology discovery and maintenance [110]. DMAC [117] is one of the first that uses the skewed scheme in which the number of active slots varies in function of the traffic load.

5.2.2.2 Semi-synchronous schemes

In semi-synchronous schemes, neighbors are grouped into synchronized clusters which interact with each other asynchronously [110]. Therefore, they mix between synchronous and asynchronous schemes. As an example, spontaneous clustering is a semi-synchronous scheme where nodes coordinate themselves without the need of a cluster-head. S-MAC ‘Sensor MAC’ [118] implements spontaneous cluster-forming scheme. In S-MAC, nodes form loosely synchronized virtual clusters by the exchange of timestamps between neighbors. Afterward, T-MAC ‘Timeout-MAC’ [119] was proposed as an improvement of S-MAC. It makes nodes switch off dynamically when the traffic activity in the neighborhood ceases. It does not only realize more energy gain but also neglects to listen to synchronization between clusters. Another semi-synchronous scheme is Elected Cluster-heads which includes the mechanisms where one of the nodes in each cluster receives the special assignment of coordinating cluster activity. For instance, LEACH [38] is a clustering-based protocol where cluster-heads are randomly rotated to guarantee energy consumption fairness.

5.2.2.3 Asynchronous schemes

In a multi-hop wireless network, since nodes synchronizing is hard and costly, asynchronous schemes were proposed. Certainly, in asynchronous schemes, nodes do not have to keep the same clock. Too many techniques incorporated in WSNs exists such as Preamble Sampling used by B-MAC [120] and WiseMAC [121]. In these types of techniques, nodes go to sleep asynchronously and wake up periodically to check for channel activity. Indeed, every frame is preceded by a long preamble longer than the duration of activity and sleep time [110]. Hence, nodes have the time to wake up, detect the transmission and stay awake to receive

the frame. Another asynchronous method is the receiver-initiated transmission where the sender waits for a periodic beacon from the receiver and transmit the frame after. RI-MAC [122] is a receiver-initiated communication protocol. In addition to these techniques, on-demand wakeup and random duty-cycling can be found. In on-demand wakeup, a node may be removed from the sleep state when necessary and it relies on a wake-up radio communication interface [110]. The random duty-cycling category is used in a dense deployment where nodes can go to sleep and wake up randomly. Hence, the duty-cycle is adjusted to the number of available nodes. RAW [123] draws on this idea.

5.3 Energy models in CCN

One of the most critical issues while enabling CCN in sensor networks is energy consumption since the medium in WSNs based on broadcast transmissions. Consequently, an important amount of interests is broadcasted on the network causing energy expiration. In this section, we present some existing energy model in content-centric networking since we propose to integrate CCN in WSNs.

In [124], the authors examined the energy consumption of content delivery architectures, taking benefits from content-centric networking. They build energy models and analyze the energy trade-off among key networking resources. They admit that the relative energy saving from CCN depends on numerous factors such as content popularity, equipment of energy efficiency, and network topology. They showed that CCN is energy efficient in popular content delivery.

While authors in [125] investigated the minimum energy consumption that CCN may achieve with optimal cache locations by considering different caching hardware technologies, a number of downloads per hour and content popularity. They also proposed a genetic algorithm to find energy-efficient cache locations. They showed that two aspects of the memory technology, energy-proportional caching and sufficient memory capacity, are critical to the overall energy efficiency gain of CCN.

Based also on the strategy of content caching, Li *et al.*, [126] studied the issue of energy-efficient caching for content-centric networking architecture. They built an energy consumption model for content distribution in an Internet Service Provider and found that the objective of energy optimization can translate into that of minimizing average response hops. They then developed an aging popularity based in-network caching where each router implements a caching policy using its local information but relying little on coordination.

To summarize, a big picture of work investigated the energy consumption for CCN, most of them are caching oriented. They showed how enhancing and proposing new content caching strategies may economize energy. Too many works also studied energy for host-centric wireless sensor networks [127, 128]. However, to the best of our knowledge, no one addressed an energy model detailing content forwarding in CCN for WSNs.

Indeed, in this chapter, to support a new forwarding strategy and to reduce the amount of energy consumption, we propose an adaptive and fully-distributed duty-cycle for content-centric wireless sensor networks. Our objective in this chapter is to address the question of how to achieve energy efficiency by switching off the radio of nodes that do not satisfy user interests (reduce the duty-cycle) in order to keep a useful backbone of nodes with a reasonable interest satisfaction rate. But first, we start by proposing a new energy model for CCN-enabled WSNs.

5.4 System Model

We consider a CCN, network with N randomly deployed nodes. Each node n can cache up to c contents in its content store (CS). Let $\mathcal{CO} = \{co_1, co_2, \dots, co_K\}$ denotes the set of generated content objects in the system and K is the cardinality of the set. The existing study proved that the request popularity distribution across different geographical locations is close to a Zipfian distribution [129] that is going to be detailed and studied in the next chapter. The choice of this distribution lay the ground to the study carried in [130] that showed that the performance of WSN with Zipf distributed nodes is improved compared to a uniform distributed nodes. Moreover, in the campus scenario, the interest distribution is more realistic when following the Zipf one. Hence, in this chapter, we assume that content popularity follows the Zipf distribution and the content objects have the same size.

We start by modeling the interest arrival process on a node and its energy consumption.

5.4.1 Proposed CCN node model

Fig. 5.3 summarizes the system model of a CCN node with the different arrival rates on the tables. The CS is the first table to be accessed when an interest packet is received. The CS hit probability denoted by p_{CS}^{hit} is the fraction of interest packets satisfied by a content object stored in the CS.

| Symbol | Description |
|-----------------------|---|
| λ_{CS}^{in} | Interests arrival rate in the CS |
| μ_{CS} | Interest rate served by CS |
| p_{CS}^{hit} | Part of Interests packet satisfied by content object stocked in the CS |
| λ_{CS}^{out} | Interest packets rate forwarded to the PIT |
| λ_{PIT}^{in} | Arrival rate of interests in the PIT |
| μ_{PIT} | Interests rate served by PIT |
| p_{PIT}^{hit} | Part of interests packet satisfied by content object stocked in the CS |
| λ_{PIT}^{out} | Interest packets rate forwarded to the FIB |
| λ_{FIB}^{in} | Interest arrival rate in the FIB |
| μ_{FIB} | Interests rate served by <i>FIB</i> |
| \bar{E}_i^n | Energy consumed by a node n for different contents |
| $E_{ca_i}^n$ | Energy consumed while caching by a node n |
| $E_{tx_i}^n$ | Energy consumed while Transmission by a node n |
| $E_{rx_i}^n$ | Energy consumed while Reception by a node n |
| $E_{p_i}^n$ | Energy consumed while checking the availability of co_i by a node n |
| co_i | Content object i |
| λ_i | Interest arrival rate in a node for a content i |
| λ_{DC} | Duty-cycle rate |
| p_i | Probability to receive an interest for a content i |
| N | Number of nodes in the network |
| K | Number of contents in the network |
| S_{co_i} | Size of content object i |
| π_{1i} | Probability of 1 st case: transmission |
| π_{2i} | Probability of 2 nd case: reception from other nodes |
| π_{3i} | Probability of 3 rd case: content unavailable |
| P_{tx} | Power density of transmission |
| P_{rx} | Power density of reception |
| t_{tx_i} | Duration of transmission |
| t_{rx_i} | Duration of reception |
| t_{ca_i} | Duration of caching |

Table 5.1. List of mathematical symbols

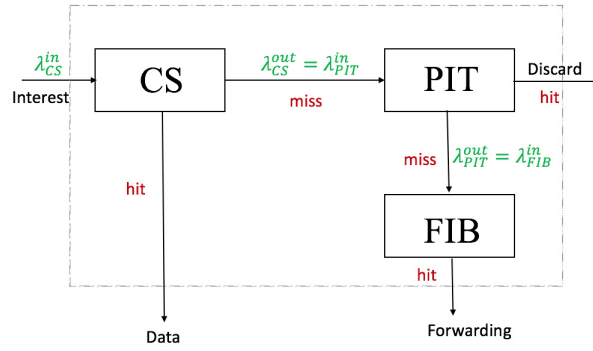


Figure 5.3. A CCN node model with three tables CS, FIB, and PIT.

The rate of interest packets that are served by the content store is:

$$\mu_{CS} = p_{CS}^{hit} \lambda_{CS}^{in} \tag{5.1}$$

If a corresponding content object is not found in the CS, the node moves to check its PIT. Thus, the rate of interest packets forwarded to the PIT is:

$$\lambda_{PIT}^{in} = \lambda_{CS}^{out} = (1 - p_{CS}^{hit}) \lambda_{CS}^{in} \tag{5.2}$$

Once the interest arrives in the PIT, its service rate is:

$$\mu_{PIT} = p_{PIT}^{hit} \lambda_{PIT}^{in} \tag{5.3}$$

Where the p_{PIT}^{hit} is the fraction of interest packets for which a request has already been issued but not satisfied. In this case, the interest is not forwarded to the FIB and an entrance is added in the PIT.

Otherwise, if a matching content is not found in the PIT, the interest is routed to the FIB table with the following rate:

$$\lambda_{FIB}^{in} = \lambda_{PIT}^{out} = (1 - p_{PIT}^{hit}) \lambda_{PIT}^{in} \tag{5.4}$$

The FIB is the last table to be reached by an interest. Then, when the interest arrives in the FIB it is served with a rate of:

$$\mu_{FIB} = \lambda_{FIB}^{in} p_{FIB}^{hit} \tag{5.5}$$

In this case, we consider that the $p_{FIB}^{hit} = 1$.

5.4.2 Proposed energy model

As already stated, broadcast is one of the most fundamental services in wireless sensor networks [112] that may cause energy wasting. The sensor nodes are tiny and battery powered devices having limited energy, hence for reliability, the foremost concern is maximizing the network lifetime while designing energy-efficient protocols and applications [131]. The energy consumption in CCN depends on several factors such as content popularity, nodes density, network topology, and nodes duty-cycle [10]. We consider that energy is dissipated while caching, checking, and especially while transmitting and receiving packets.

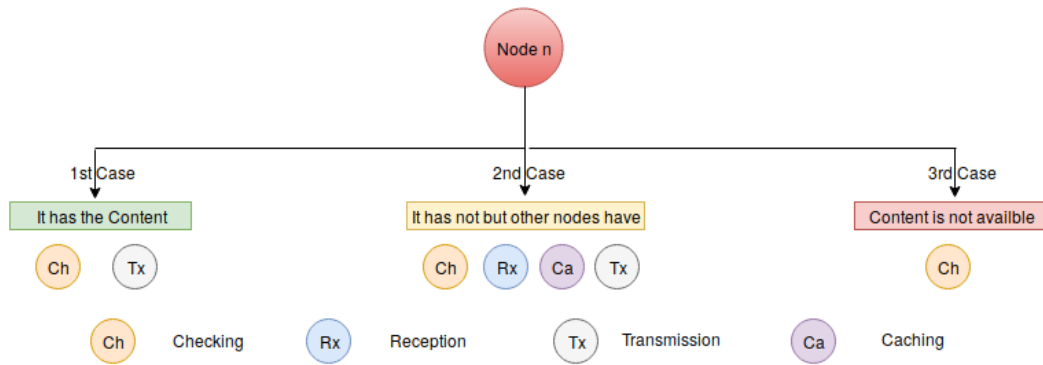


Figure 5.4. Energy consuming operations.

We have three cases detailed in Fig. 5.4. In the first one, when a node receives an interest for a content object i (co_i), it checks the availability of this content object. If the node finds co_i in its CS, it sends the content back to the requester. In the second case, the node does not have the co_i , it receives the content from other nodes, it caches it in its content store, and it forwards it. In the last case, the content co_i is not available at any node in the network. Note that the proposed energy model does not take into consideration certain basic energy consumption but focuses on important energy consumption per case.

In our model, the energy spent depends on the node status. Then, on its duty-cycle rate λ_{DC} . We will detail the integration and the dependence on the duty-cycle in the next section. Consequently, the energy consumed by a node n in a network with C requested contents, given that an interest has been received, is:

$$\bar{E}_i^n = \lambda_{DC} \sum_{i=1}^K (\pi_{1i} E_{1i}^n + \pi_{2i} E_{2i}^n + \pi_{3i} E_{3i}^n) \quad (5.6)$$

where E_{1i}^n , E_{2i}^n , E_{3i}^n denote the energy consumption during the first, the second, and the third case respectively. Let π_{1i} , π_{2i} , π_{3i} denote the probability to be in the first, the second, and the third case respectively when a request for content i is received. λ_{DC} is the duty-cycle rate and K is the number of contents.

Case 1:

When receiving an interest for a co_i , the node starts by checking the availability of this content in its CS. If co_i is available, the node transmits the corresponding content. In this case, the energy used is equal to:

$$E_{1i}^n = E_{tx_i}^n + E_{p_i}^n \quad (5.7)$$

where $E_{tx_i}^n$ is the energy used by a node for the transmission of a CO and $E_{p_i}^n$ is the energy used for checking the availability of the content in the CS.

We start with the energy consumed while transmitting a data packet. The latter is expressed as follows:

$$E_{tx_i}^n = \alpha t_{tx_i} P_{tx} \quad (5.8)$$

The equation includes a factor α used to count for redundancy and overhead caused by the transmission [132], the transmission time t_{tx_i} , and the node transmission power P_{tx} (mW).

In CCN, the transmission of a content object occurs if the node receives an interest for the content co_i giving that it received an interest and if the content is available in its CS. Thus, the probability of transmission for content co_i is:

$$\pi_{1i} = p_i p_{CS_i}^{hit} \quad (5.9)$$

where p_i denotes the probability to receive an interest for the i^{th} type of content and $p_{CS_i}^{hit}$ is the probability of success to find it in the CS.

Since the interests follow the Zipf distribution [129], Zipf's law states that the i^{th} most popular content requested with a probability p_i depends on the request rate of this content λ_i^n :

$$p_i = \lambda_i^n / \sum_{j=0}^K \lambda_j^n \quad (5.10)$$

K is the number of generated contents in the network.

λ_i^n represents the request rate for content types co_i arriving at a node n from a node l where $l \in S^n$; S^n denotes the set of neighbors :

$$\lambda_i^n = \sum_{l \in S^n} \lambda_i^l (1 - p_{CS_i}^{hit}(i, l)) \quad (5.11)$$

We admit that the aggregated requests rates follow also the Zipf distribution [133].

The hit probability of a content i taken from a set of K contents [133] is equal to:

$$p_{CS_i}^{hit} = 1 - e^{-p_i \tau_i} \quad (5.12)$$

Where τ_i is the time between two adjacents satisfied interests for content i . τ_i is a constant for any given co_i [133]. This rate becomes more and more deterministic when the aggregated arrival rate of all other content interests increases.

Thus the probability related to the 1st case is:

$$\pi_{1i} = p_i (1 - e^{-p_i \tau_i}) \quad (5.13)$$

Finally, we deduce the expression of Eq. (5.7):

$$E_{1i}^n = \alpha t_{tx_i} P_{tx} + E_{p_i}^n \quad (5.14)$$

Case 2:

Once a node sends the content object, the nodes that solicited the co_i before, receives it, caches it in their CS and forwards it. During these operations, the energy consumption can be written as:

$$E_{2i}^n = E_{rx_i}^n + E_{ca_i}^n + E_{tx_i}^n + E_{p_i}^n \quad (5.15)$$

Let $E_{rx_i}^n$ be the energy spent for content receiving and $E_{ca_i}^n$ be the energy spent for caching.

The node receives the co_i if it requested it earlier and if it does not have the content in its CS. Thus, the

probability to be in that case is expressed by:

$$\begin{aligned}\pi_{2i} &= p_i (1 - p_{CS_i}^{hit}) \\ &= p_i e^{-p_i \tau_i}\end{aligned}\quad (5.16)$$

The energy spent for receiving is:

$$E_{rx_i}^n = t_{rx_i} P_{rx} \quad (5.17)$$

Where t_{rx_i} and P_{rx} represent respectively the duration and the power of reception (mW).

Finally, the node checks its CS, if it does not have the co_i , it caches it and forwards it. Assuming that the size of a content co_i energy is S_{co_i} , if a copy of content co_i is cached for a duration t_{ca_i} , the energy consumed by the node is equal to:

$$E_{ca_i}^n = S_{co_i} t_{ca_i} P_{ca} \quad (5.18)$$

P_{ca} denotes the power density of caching in a node.

Hence, the energy dissipated is:

$$E_{2i}^n = t_{rx_i} P_{rx} + S_{co_i} t_{ca_i} P_{ca} + t_{tx_i} P_{tx} + E_{p_i}^n \quad (5.19)$$

Case 3:

The requested content co_i can not be proceeded by any node in the network which means that the content is not available.

The probability depends on the probability of reception of an interest for this content and on the probability of not having this co_i in the CS, Then:

$$\begin{aligned}\pi_{3i} &= 1 - \pi_{1i} - \pi_{2i} \\ &= 1 - p_i\end{aligned}\quad (5.20)$$

The energy used in this case is only for checking:

$$E_{3_i}^n = E_{p_i}^n \quad (5.21)$$

By combining Eq.(5.14), Eq.(5.19), and Eq.(5.21), the total energy consumed by each node is written as:

$$\begin{aligned} \bar{E}_i^n = \lambda_{DC} \sum_{i=1}^K & \left(p_i (1 - e^{-p_i \tau_i}) (\alpha t_{tx_i} P_{tx} + E_{p_i}^n) \right. \\ & \left. + p_i e^{-p_i \tau_i} (t_{rx_i} P_{rx} + S_{co_i} t_{ca_i} P_{ca} + t_{tx_i} P_{tx} + E_{p_i}^n) + (1 - p_i) E_{p_i}^n \right) \end{aligned} \quad (5.22)$$

5.4.3 Numerical analysis of the energy model

In this part, we study the variation of energy consumption for one node requesting one content by varying different parameters in order to analyze their impacts. The sensor characteristics given in Table 5.2 are taken from the specifications for the Tmote-Sky sensors [134].

| Parameters | Value |
|-----------------------|---------|
| Transmit (P_{tx}) | 58.5 mW |
| Receive (P_{rx}) | 65.4 mW |
| Caching (P_{ca}) | 15 mW |
| Checking (P_p) | 15 mW |
| t_{tx} | 4.25 ms |
| t_{rx} | 4.25 ms |
| t_{ca} | 0.5 ms |
| τ_i | 100 ms. |
| α | 2 |

Table 5.2. Analysis parameters

For the implementation, we built our energy model into a custom MATLAB simulator and we try to explain how energy behaves when varying the duty-cycle, the rate of interests, and τ_i . Based on the values given in Table 5.2, we plot the energy consumed by a node for one content object delivery.

Figure. 5.5 plots the energy consumption for different p_i and λ_{DC} values. For $\tau_i = 0.1$ s, it is worth noting that, when the duty-cycle is low, the energy consumption is low compared to a high duty-cycle (100 %). As expected, when the probability of reception of an interest for a content co_i increases, the

energy consumption also increases since the node dissipates energy while checking co_i , transmitting the corresponding content (if cached in CS), or receiving it from a neighbor.

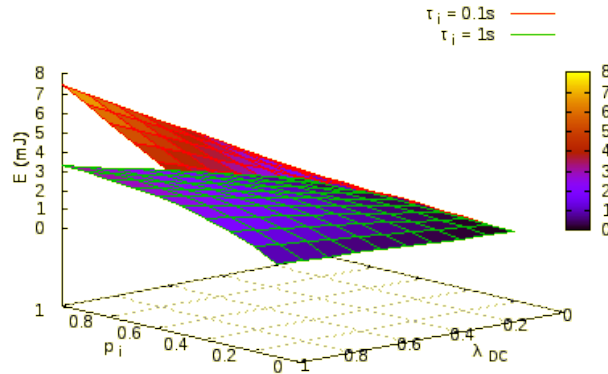


Figure 5.5. Energy consumption vs. p_i , λ_{DC} , and τ_i .

We also evaluate the impact of τ_i on the energy consumption. The energy cost of a content delivery is plotted in Fig. 5.5 comparing also two values of τ_i . An interesting observation comes from Fig. 5.5, when $\tau_i = 100 ms$ and $p_i = 1$, the energy is less than when $\tau_i = 1000 ms$. Hence, energy consumption for the first case is nearly equal to $8 mW \times s$. Whereas, when the interest inter-arrival time $\tau_i = 1000 ms$, the energy is about $3 mW \times s$. Consequently, when τ_i increases, the energy consumption decreases. This could be explained by the fact that since τ_i denotes the inter-arrival time between two interests without a cache miss. When it increases, it means that the content is not available. So, the nodes save their energy.

5.5 Proposed duty-cycle plan

In low duty-cycle WSNs, nodes stay in a dormant state most of the time to save energy and wake-up for a very short period of time, which poses challenges to a reliable forwarding [16]. This is particularly true for CCN because of the need to properly forward Interest and Content Object messages. To this end, we propose a mechanism that takes into account the CCN features and achieves energy efficiency at the same time.

5.5.1 Principle

As aforementioned, too many duty-cycle plans were proposed in the literature (discussed in section 5.2). A duty-cycle plan presents a very effective way to save energy by putting the radio of nodes in standby mode (low-power) whenever possible [135]. Radio energy consumption contributes to the overall energy

consumption at each node. As previously mentioned, duty-cycling schemes are usually classified into synchronous and asynchronous in relation to the mechanism used to coordinate the schedules of the nodes [110]. The proposed duty-cycling mechanism is asynchronous since when a node realizes that it is not involved in the interest satisfaction (i.e., their PIT contain too many unsatisfied content requests), it reduces its duty-cycle by switching off its radio for a defined time interval. According to the CCN standard implementation, the PIT table contains a list of unsatisfied interests. Therefore, the PIT size increases, when the node is not involved in satisfying the user interests.

In other words, in order to save energy, we propose to reduce the duty-cycle of the nodes that have a percentage of entrances in their PIT (respecting to the PIT size and the total number of received interests) greater than a given threshold. Thereafter, when an entry associated with an interest in the PIT is deleted as soon as a new content object is sent back to the user, the duty-cycle of the node is increased by a percentage. In the proposed algorithm, the node measures periodically the percentage of entrances in its PIT (n_p). When this percentage exceeds a certain threshold, the node decides to reduce its duty-cycle.

The choice of the *threshold* and the percentage by which the duty-cycle is increased/decreased can be tuned based on the scenario. The parameter sensitivity of the algorithm is investigated in the next section through extensive simulations.

Let denote P , the percentage by which the duty-cycle is increased/decreased. P depends on the threshold by which the node starts reducing its duty-cycle. Once the size of the PIT increases with a certain threshold, the node starts reducing its duty-cycle in function of this threshold and the PIT maximum size. If the threshold is not too high which means that the node does not have a lot of unsatisfied interests, the node does not decrease a lot its duty-cycle. Otherwise, when the threshold is big, this means that the node has a lot of unsatisfied interests in its PIT. Then, the node is not participating in replying with requested contents so in increasing the user satisfaction. Therefore, the node will reduce its activity by a bigger percentage. Then, the percentage P is a correlation between the threshold and the maximum size of the PIT. The node must not turn 100% on a sleep mode because the mechanism has to ensure a certain degree of activity (larger than zero) to ensure a minimum level of Quality of Service (QoS). Since P depends on the threshold the node will never turn into sleep mode for 100%.

In ADDC-CCWSN, we assume a WSN consisting of a number of sensor nodes which is deployed in a

Algorithm 2: ADDC-CCWSN

```

input   :  $n_p$ : Percentage of entrance in PIT
output  :  $\lambda_{DC}$ : Duty-cycle frequency
           :  $P$ : Percentage of activity time reduction

//If the entrances in PIT are bigger than a threshold, the node reduces its activity by
//  $P$  % since it is not realizing user interest satisfaction.
1:  $\lambda_{DC}=1$ ;
2: while receive Interests do
3:   if ( $n_p \geq threshold$ ) then
4:     Calculate  $P = \frac{Threshold}{max\_PIT\_size}$ ;
5:     Reduce the activity time by  $P$  %;
6:      $i = 1$ ;
7:     //Decrease  $\lambda_{DC}$ 
8:      $\lambda_{DC} = \lambda_{DC} - \frac{P*i}{100}$ ;
9:      $i = i + 1$ ;
10:    Update  $n_p$ ;
11:    if ( $n_p < threshold$ ) then
12:       $i = i - 1$ ;
13:      Increase the activity time by  $P$  %;
14:      //Increase  $\lambda_{DC}$ 
15:       $\lambda_{DC} = \lambda_{DC} + \frac{P*i}{100}$ ;

```

given field. Each node is duty-cycled with two possible states, the active state and the doze state switched according to a working schedule.

5.6 Performance Analysis

In this section, we examine the performance of the proposed duty-cycle mechanism to evaluate its impact on the energy consumption and the interest satisfaction rate based on the equations derived in section 5.4.1. Hence, the performance of our approach is evaluated under various conditions and compared to an approach where the network nodes are 100% awake (mostly-on nodes).

5.6.1 Simulation set-up

For the implementation of the proposed scheme, we continue with CCNx_Contiki framework [70] and we extend it to follow the requirements of our mechanism. The simulation is conducted using Cooja simulator.

| Simulation parameters | Value |
|------------------------------|-------------------|
| Number of nodes | 80, 160, 240 |
| Number of entrance points | 3 |
| Wireless interface | IEEE 802.15.4 |
| Radio coverage range | 100 <i>m</i> |
| Initial energy | 2 <i>J</i> |
| Interest size | 25 <i>bytes</i> |
| Content Object size | 75 <i>bytes</i> . |

Table 5.3. Simulation parameters

The simulations run 10 times using different random seeds and present the mean values as results. 95% confidence intervals (CIs) are calculated and the values are below 8.5×10^{-4} . Therefore, confidence intervals are not drawn on the figures since they are very low.

Let consider a wireless sensor network deployed on an area of $500\text{ m} \times 500\text{ m}$ corresponding to a small part of the campus university described in the application scenario. The number of nodes varies from 80 to 240. All the sensor nodes are static and have the same detection radius. In addition, all the communication links are bidirectional. Different types of content are generated in the network. These interests are injected through entrance points. The framework imposes that interests must penetrate in the network through an entrance point. In other words, entrance points in this thesis represent points on which interests are injected into the network. Behind these 3 points, we suppose that there is a population of users who broadcast their interests. For instance, behind entrance point 1, there exist 30 users, behind entrance point 2, 90 users and behind entrance point 3, 30 users. Entrance points realize aggregation of certain interests asking for the same contents at the same time. Simulation parameters are detailed in Table 5.3. For the sake of illustration, we take into account not only the energy consumption modeled in the previous section but also the energy consumed due to the interests forwarding. To evaluate the energy consumption, we adopt the values cited in Table 5.2.

5.6.2 Evaluation metrics

In this chapter, in order to validate our contribution, for the performance evaluation, we introduce certain metrics.

As a first and important metric, we choose the energy consumption which represents the total energy consumed by the network presented in the proposed energy model. The energy model does not take into consideration basic energy consumption in WSNs but it is based on the considered cases in CCN architecture detailed also in Fig. 5.4.

We also propose to evaluate the interest satisfaction rate presented by the ratio between the number of satisfied and generated interests.

$$Interest_satisfaction_rate = \frac{\sum_{i=1}^C Satisfied_interests}{\sum_{j=1}^N generated_interests} \quad (5.23)$$

Satisfied interests are interests replied with C content objects and generated interests are the total of all generated interests for N contents in the network.

We propose also to evaluate the delay. In other words, the average time between the broadcast of the interest by the user and the recovery of the corresponding content object.

And finally, we present results for the packet loss ratio known as packet lost due to overmitting or congestion in the network.

5.6.3 Simulation results

The evaluation results are presented under two different values of thresholds. Indeed, the nodes start reducing their duty-cycle when the percentage of entrance in their PIT (respecting the PIT size and total the number of received interests) exceeds a given threshold compared to its capacity. Hereinafter, we study the impact of this threshold.

5.6.3.1 Study of the threshold impact

The choice of reducing the duty-cycle of the nodes is related to the percentage of unsatisfied interests that the node has in its PIT. In this section, we try to investigate the best threshold to start reducing the node duty-cycle. A threshold that reduces energy without impairing the interest satisfaction rate.

Simulation runs for two thresholds, 20 % and 50 % of unsatisfied interests in PIT.

Simulation 1: In this simulation, the threshold is fixed to 20 % of unsatisfied interests in the PIT. If the percentage of entrances in the PIT exceeds 20 %, the node starts reducing its duty-cycle.

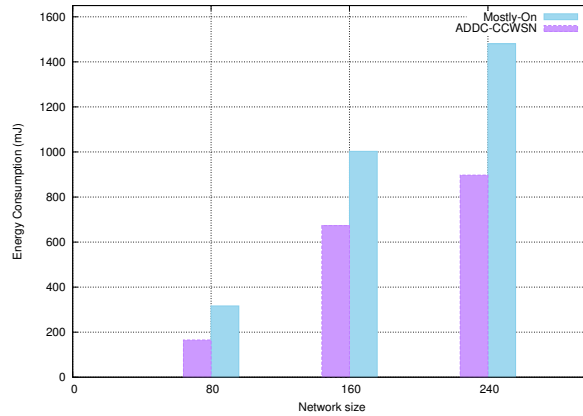


Figure 5.6. Energy consumption vs. network size for a threshold 20%.

Fig. 5.6 shows the energy consumption for different network sizes when the PIT contains 20% of unsatisfied interests. As depicted in Fig. 5.6, as the network density increases from 80 to 240, the energy consumption increases. Indeed, more nodes receive and forward packets due to the interest flooding in the wireless network. It is observed that with mostly-on nodes, for 160 nodes, the network consumes about 1000 *mJ* of energy while with ADDC-CCWSN, the network consumes just 673 *mJ*. Then, our scheme realizes a gain of 35 % in terms of energy consumption. Absolutely, in the proposed approach, when nodes realize that they are not involved in the interest satisfaction, they reduce their duty-cycle and switch to doze state for a defined time interval. Thus, they economize energy. It is also noticed that when the network density increases, the energy saved by ADDC-CCWSN increases since there are more nodes that reduce their duty-cycle. For 240 nodes, a gain of 42 % of energy consumption is realized compared to the mostly-on scheme.

In this scenario, when the nodes notice that the percentage of entrances in their PIT (respecting the PIT size and the total number of received interests) exceeds 20%, they reduce their duty-cycle. Figure. 5.7 illustrates the interest satisfaction rate for different network size under a threshold of 20 %. For 80 nodes, the mostly-on scenario realizes a satisfaction rate of 86 % while ADDC-CCWSN realizes 72 %.

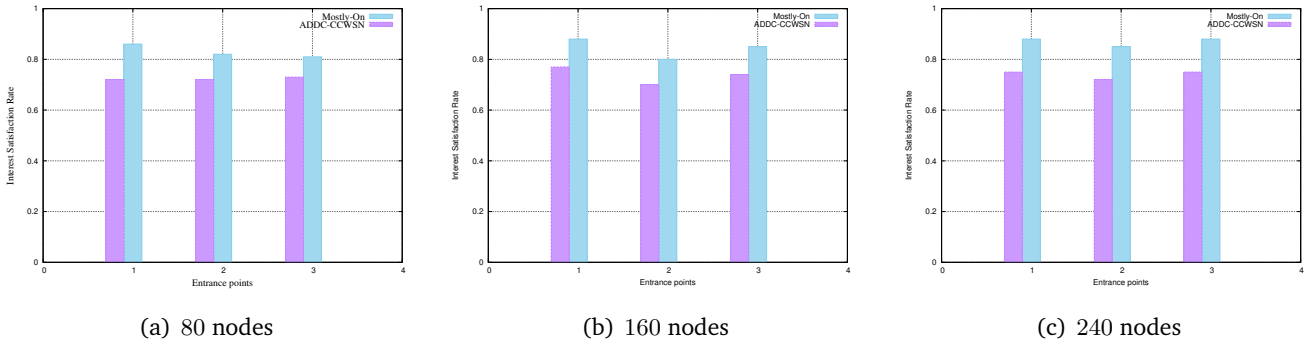


Figure 5.7. Interest satisfaction rate for different network sizes with a threshold equal to 20%

So, the rate decreases by 14 %. For the mostly-on approach, the rate is not of 100 % since the interest may be dropped due to congestion. When the duty-cycle is decreased, there is a risk that the node misses interests or contents forwarding. Then, this may impact the interest satisfaction rate which explains the results.

Reducing the activity of a node when it realizes that it has 20 % of unsatisfied interests in its PIT realizes energy saving while decreasing the interest satisfaction rate under the presented conditions and the given parameters.

Simulation 2: In this simulation, the threshold is fixed to 50 %. If the percentage of unsatisfied interests in PIT exceeds 50 %, the node starts reducing its duty-cycle by a step of 10 %. For example, if the percentage of unsatisfied interests in PIT became 70 %, the node reduces its duty-cycle by 30 %.

Figure 5.8 demonstrates a gap in terms of energy consumption between mostly-on and the proposed adaptive duty-cycle scheme. It is worth noting that for 80 nodes, when the nodes are mostly-on, they consume 316 *mJ* while when they reduced their duty-cycle (they applied ADDC-CCWSN), they just consumed 226 *mJ*. So they realized a gain of 28 %. Thus, the proposed scheme effectively enables energy saving. Furthermore, when the network density increases, the energy consumption increases. Indeed, in content-centric wireless sensor networks, more nodes receive the flooded interests and try to reply with a corresponding content. For the adaptive duty-cycle mechanism, when the number of nodes increases, the gain in terms of energy increases. For example for 240 nodes, the proposed ADDC-CCWSN realizes a gain of

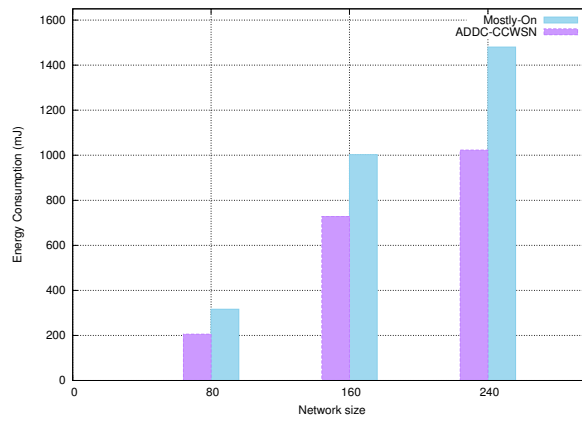
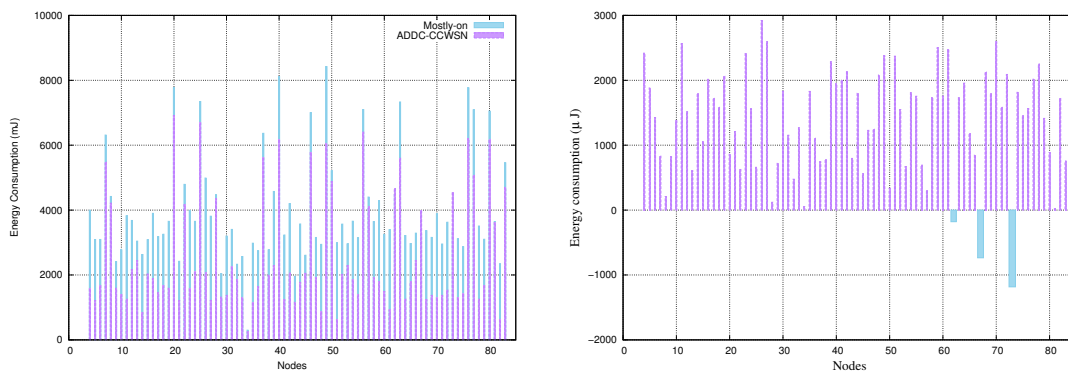


Figure 5.8. Energy consumption for threshold 50% vs. network size.

36 %. Certainly, more nodes reduce their duty-cycle and minimize energy consumption.



(a) Energy consumption for each node.

(b) Energy saving or wasting for each node.

Figure 5.9. Energy consumption for 80 nodes and the threshold 50%.

Let detail the simulation results for 80 nodes. As shown in Fig. 5.9(a), practically 95 % of the nodes realize a gain. Some nodes consume the same amount of energy for the mostly-on and ADDC-CCWSN, which means that they did not reduce their duty-cycle. An interesting observation comes from the Figure. 5.9(b), the nodes 62, 67 and 73 consume more energy in the adaptive scheme than mostly-on because their neighbors were sleeping while they received interests. Consequently, those nodes handle the forwarding of

interests and contents object and consume more energy as a result.

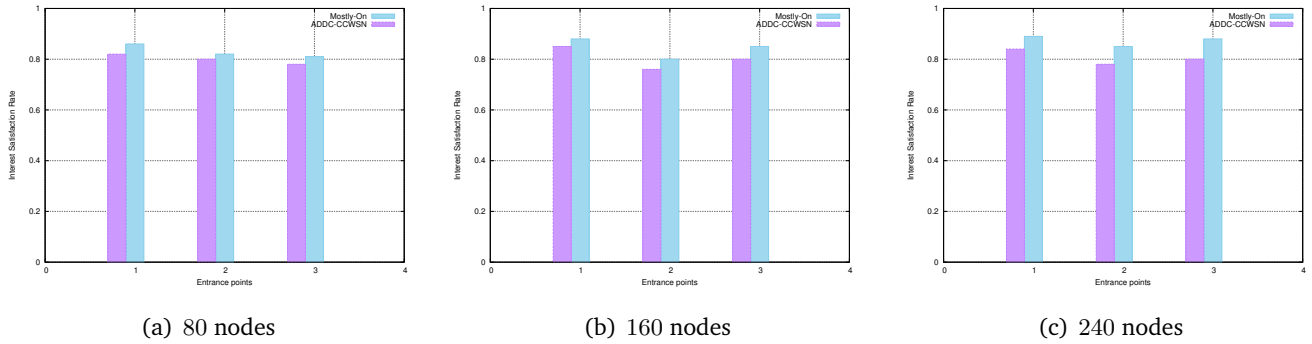


Figure 5.10. Interest satisfaction rate for threshold 50% and different network sizes.

While this part has so far explored the energy consumption results for the threshold 50 %, we now briefly examine the interest satisfaction rate implications for the same threshold.

Figure 5.10 plots the interest satisfaction rate for different network sizes for three entrance points under mostly-on and ADDC-CCWSN schemes. We note that the proposed scheme slightly impact the interest satisfaction rate. As illustrated in Fig. 5.10, for 80 nodes, the interests sent by user 1 (behind entrance point 1) were satisfied with a rate of 86 % for mostly-on approach and 82 % for the proposed approach. Then, ADDC-CCWSN just realizes a lost of 4 %. Therefore, reducing the duty-cycle of certain nodes in the network (when exceeding the threshold 50 %) does not have a significant impact on the interest satisfaction rate.

Fig 5.11 shows the average delay behind different entrance points. The delay trend among ADDC-CCWSN is similar to the mostly-on scheme. Users wait for 0.3 s to get the content. Thus, without consideration of collisions and a none response with a content, the duty-cycle control does not impact the propagation delay of data. Certainly, when the nodes reduce their duty-cycle, their neighbors take over.

Finally, we investigate the packet loss ratio in the network for both thresholds 20 % and 50 %.

As depicted in Fig. 5.12, the packet loss ratio is more important for the threshold 20 %. This is due to the doze state of nodes. As a node notices that the entrances exceed 20 % of unsatisfied interests in PIT, it reduces its duty-cycle. However, for the second threshold, the node waits till 50 % of unsatisfied interests

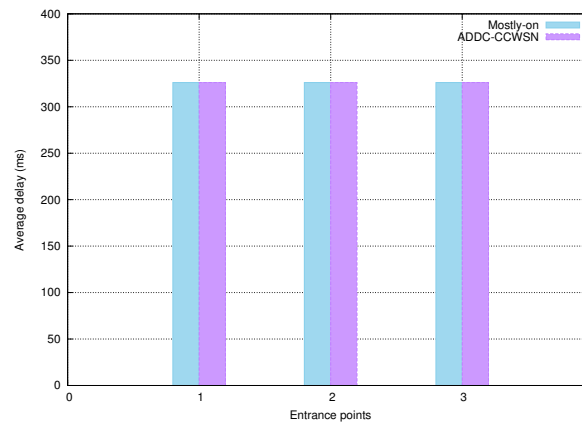


Figure 5.11. Mean delay realized by different schemes for both thresholds.

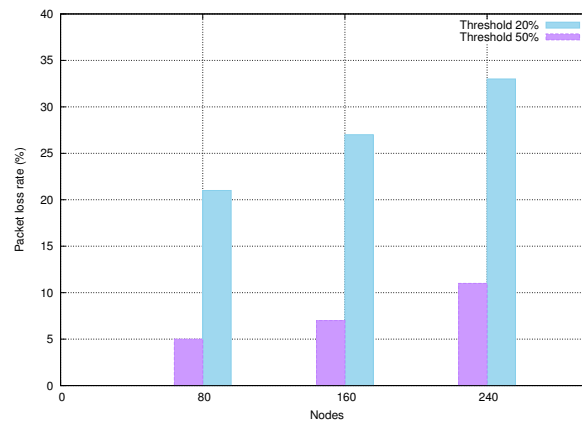


Figure 5.12. Packet loss ratio for both thresholds 20 % and 50 %.

in PIT to reduce its duty-cycle. For instance, in case of 80 nodes, when the threshold is equal to 50 %, the packet loss ratio is equal to 7 % while for 20 % the packet loss ratio is equal to 21 %. This highlights why threshold 50 % realizes better interest satisfaction rate. It is worth to note that between the 21 % lost packets, we find duplicated and re-forwarded interests because of flooding and cache misses. Indeed, the interest satisfaction rate shown in Fig. 5.7 for the threshold 20 % endorses this.

5.6.4 Discussion

The comparative threshold performance analysis has shown that the threshold 20 % realizes more energy saving than the threshold 50 %. Since the nodes decide to reduce their activity earlier, they realize more

energy gain. However, there is not a significant difference between the two thresholds. For example, for 80 nodes, the threshold 20 % realizes a gain of 32 % while the threshold 50 % saves 28 % of energy. Otherwise, even if the threshold 50 % seems late to save more energy, it does not significantly decrease the interest satisfaction rate. These results lay the groundwork for waiting until having 50 % of unsatisfied interest in their PIT to start reducing their duty-cycle. The nodes may miss interests or contents forwarding. However, for the threshold 20%, where the nodes decide earlier, they miss more interests and contents forwarding. Which explains why the threshold 50 % achieves a better interest satisfaction rate. In addition, a close look into the interest satisfaction rate reveals that this latter is better for all the users behind the different entrance points (for both thresholds). This is because nodes connected to point 1 and point 2 have a higher number of neighbors. This effect is purely dependent on the number of neighbors. Absolutely, the higher the number of neighbors, the higher the probability that some nodes will be active when necessary (they will forward the interests and reply with the content when their neighbors are sleeping).

The results have also highlighted that for the delay, reducing the duty-cycle do not impact the delay since the neighbors of a sleeping node take over and forward the interests or the content object.

Our analysis for the packet loss indicated that the threshold 50 % gives a better ratio than the threshold 20 %. In particular, when the nodes decide to reduce their duty-cycle earlier, they miss more interests and contents forwarding. However, it is worth to note that the nodes may miss flooded and duplicated interests (generated due to content miss) which can not impact the result of the interest satisfaction rate. This is the reason behind having an important packet loss ratio while realizing a reasonable interest satisfaction rate.

The threshold 50 % achieves better results than the threshold 20 % in terms of interest satisfaction rate and packet loss ratio. Moreover, there was not a huge difference in terms of energy saving compared to the threshold 20 %.

The results in this section have shown so far that the threshold 50 % is better but we argue that the choice of the threshold and the duty-cycle reducing differs from a scenario to another and depends on the scenario parameters. Furthermore, it was shown that it would be better to find a trade-off between energy consumption and the interest satisfaction rate.

5.7 Conclusion

Duty-cycling schemes are widely used for energy conservation in wireless sensor networks. This type

of schemes allows sensor nodes to achieve long-run energy neutrality. For this reason, in this chapter, we investigate the energy consumption that CCN achieves with an optimal forwarding scheme. Thus, a forwarding mechanism is presented by a duty-cycle plan for energy efficiency in content-centric WSNs (ADDC-CCWSN). The proposed scheme evokes the activity reducing of the nodes considered as useless in the network while keeping a stable backbone of nodes (with high-interest satisfaction rate). Therefore, nodes start reducing their duty-cycle when the percentage of entrances in their Pending Interest Table (PIT) exceeds a certain threshold. To build the scheme, we derive the CCN node operating and energy consumption models. Unlike other energy models, our proposed model is a CCN architecture based and takes into consideration the node duty-cycle and the forwarding scheme in WSN. The purpose was to conduct a comparative study between the mostly-on approach and the newly proposed scheme ADDC-CCWSN across two thresholds. Finally, we present simulation results to highlight the impact of ADDC-CCWSN on energy consumption and interest satisfaction rate. The results have shown so far that a threshold may realize better results than another. In fact, our goal was not to look for the best threshold, we realized a study based on two threshold values and we noticed that reducing the duty-cycle earlier saves energy but reduces significantly the interest satisfaction rate. Therefore, our main goal is to reduce the duty-cycle of nodes that do not participate in replying by requested contents while keeping a backbone of nodes that ensures this. In addition to that, we wanted to achieve an enhancement in terms of energy consumption while keeping a high-interest satisfaction rate.

CHAPITRE 6

Content Popularity Aware Caching Strategies in WSNs

Contents

| | | |
|------------|--|------------|
| 6.1 | Introduction | 92 |
| 6.2 | State of the art | 92 |
| 6.2.1 | Data caching in WSNs | 92 |
| 6.2.2 | Content caching in CCN | 93 |
| 6.3 | A collaborative caching strategy for content-centric enabled wireless sensor networks | 98 |
| 6.3.1 | Content request process | 98 |
| 6.3.2 | Cache admission control | 99 |
| 6.3.3 | Cache replacement policy | 102 |
| 6.4 | Performance evaluation | 104 |
| 6.4.1 | Simulation set-up | 104 |
| 6.4.2 | Evaluation metrics | 105 |
| 6.4.3 | Simulation results | 106 |
| 6.4.4 | Discussion | 116 |
| 6.5 | Conclusion | 117 |

6.1 Introduction

Wireless sensor networks are usually deployed to monitor large areas, collecting data with regular frequency. This large volume of data has to be stored somewhere for answering to external user queries [136]. Generally, data transmission in WSN consumes more energy than processing, then it is good to use the benefits of caching so that data access can be done faster [137]. Caching if used efficiently, could reduce overall network traffic and hence optimize bandwidth utilization. Caching data at locations that minimize packet transmissions in the network reduces the power consumption in the network, and hence extends its lifetime.

Initially, in WSNs no in-network storage was considered: the request for data was routed from the sink to every sensor by flooding messages [136]. Data were sent back to the sink by following the same path but in the reverse direction. Moreover, a lot of constraints exist in WSNs. To overcome such constraints, cooperative caching techniques are used. If they are efficiently used then the network lifetime can be enhanced and data loss may also reduce during transmission and reception process.

Caching is the essential benefit of content-centric networking which has been used for enhancements like fault tolerance, improving communication over wireless sensor network, multi-casting applications, and improving the network performances [138].

In this chapter, we show the benefit of caching in content-centric wireless sensor networks. Furthermore, we propose a new collaborative caching strategy and we study the existing approaches in order to compare them with our strategy.

6.2 State of the art

6.2.1 Data caching in WSNs

The proliferation of sensor applications where a huge amount of data is generated at active nodes and sink depends mainly on the ability of protocols to support a large number of sensors, to save energy and to provide answers in short delays. Cooperative data caching has been proposed as an effective and efficient technique to achieve these goals concurrently [139]. It uses a cache discovery process, cache consistency and cache replacement policies for the enhancement of the network lifetime. In addition to that, in this type of protocols, the most important thing is the selection of the sensor nodes which will take special roles in running the caching and requesting forwarding decisions. It was also demonstrated in [140] that caching

the useful data for each sensor either in its local storage or in the neighborhood nodes can increase the network lifetime. Moreover, caching if optimally implemented can reduce significantly network traffic and helps in providing higher data availability to the users (sink) in wireless sensor networks.

Content-Centric Networking affords efficient caching capabilities. As mentioned earlier, CCN caching ensures high content availability, network traffic reduction, and low retrieval latency which reduces congestion and improve end-to-end delay. To this end, we propose to exploit this characteristic and enable CCN in WSNs environments. The structure of CCN mechanism enables the storage of data in every sensor in wireless sensor networks. Too many research works have been devoted to the cache management problem in the caching nodes of sensor networks. However, caching the content on all the nodes of the networks is not a good strategy in terms of resource utilization. So where to cache and how to cache in order to optimize the resources while realizing a high-interest satisfaction rate? This is our purpose. We aim to study existing caching strategies and propose a strategy that may compete the existing ones.

Several works have been proposed by the authors exploiting caching the data either in some intermediate nodes or at a location nearer to the sink in wireless sensor networks [75]. Indeed providing solutions to optimally caching the data has been a big area to be focused on. For instance, authors [78] proposed a collaborative caching strategy for information-centric wireless sensor network (ICN-WSN). The proposed strategy is based on the node betweenness based cache size adjustment, the data replacement frequency based cache decision, and the content value based cache replacement algorithm.

To summarize, many works investigated content caching in CCN. Several works treated this in CCN-WSNs. However, to the best of our knowledge, no one proposed a caching strategy that combines at the same time the node degree and its distance from the source, and aims to reduce energy consumption, traveled path and increase cache diversity. Motivated by the aforementioned shortcomings, this chapter presents a study of existing caching strategy applied in WSNs and proposes a design of a caching strategy to decide where to place content copies depending on the node degree and its distance from the source.

6.2.2 Content caching in CCN

In-network caching in CCN can present some decision challenges: the cache placement (where to cache), content replacement (which content is to evict from the cache) and request routing (how to redirect requests to optimal cache). More specifically, optimal cache placement is strictly dependent on content admission

control, request routing, network topology, content replacement policy, interest arrival rate, and caching size [51].

6.2.2.1 Interest routing in CCN

Interest routing can be divided into on-path and off-path routing generating an on-path or off-path content caching.

On-path: With on-path request routing, interests are first routed from the requester to the closest cache. Then, they are routed over the network of caches towards content origin using shortest path routing and are served from a cache only if the requested content is available at a node on the request path [141]. This technique is suitable for edge caching strategies since interests are routed to the closest cache but in case of a cache miss, they are forwarded directly to source nodes.

Off-path: Off-path routing can be implemented using a centralized (with a global view of cached contents queried before routing an interest) or distributed content-to-cache resolution process. This approach is suitable for systems operating under proactive and reactive content placement as long as content location do not vary frequently. For reactive caching systems with a high rate of content replacement, a number of more scalable off-path interest routing algorithms have been proposed with an objective to enable caching nodes to exchange state among each other in a lightweight manner and route requests accordingly [141].

Once the interest routing is established and a content is found on the path, the next problem is how and where to cache this content? The most simple content placement strategy is to leave a copy of the content in every node gone through, which is known as Leave Copy Everywhere (LCE). However, this strategy causes a high degree of redundancy as all caches along the delivery path consume cache resources to hold identical items. To reduce this redundancy and, therefore, to increase the diversity of cached contents along a delivery path, various algorithms have been proposed to select nodes on which caching is done. These algorithms include Leave Copy Down (LCD) [142], Cache less for more [143] and ProbCache [144].

6.2.2.2 Cache placement strategies

Caching data at locations that minimize transmissions in the network reduces the power consumption in the network, and hence extends its lifetime. However, excessive caching can lead to high costs and performance

degradation. Finding the best locations of the nodes for caching optimizes the communication [145]. The objective of the chapter is to study existing caching strategies but this time applied in WSNs and propose a competitive one. The essence of the proposed mechanism is the selection of the sensor nodes which will take special roles in running the caching and request forwarding decisions.

Leave Copy Everywhere cache management in LCE [142] is defined by its operation of caching data in every node crossed as shown in Fig. 6.1(a). Part of the practice of the ICN is the ability to make information readable and easily accessible as described in ICN initial proposal. As a user sends out a request using LCE, the nature of the network serves the interest using a hierarchical search and ordering of nodes to acquire a cache-hit. LCE is generally a good choice in case of flash-crowd events or in case of highly skewed content popularity distributions and it does not require any coordination [141].

Leave Copy Down the Leave Copy Down (LCD) [142] is a cache management strategy that defines the form and manner of content caching on nodes. As depicted in Fig. 6.1(b), its operation works in a similar fashion to the popular ‘drop at the first neighbor’ process. This technique requires minimal coordination among caching nodes as they can signal to other nodes downstream whether to cache the content or not by simply appending a flag to the delivered content. [141].

Betweenness Centrality centrality is a measure of the importance of a node in the communication model. In this algorithm [143], the centrality of the node is used as a parameter to select the appropriate node. The more number of times a node will come on the content delivery path, the more centrality of the node will be high. BetwCent chooses a node with the highest betweenness centrality along the content delivery path to store contents which improves the caching and eliminate the uncertainty in the performance of the simplistic random caching strategy. The mechanism of this policy is presented in Fig. 6.1(c).

Pop-cache, cache less or more based on content popularity the popularity of contents has been taken as deciding factor in this algorithm [144] because more popular content should be cached on the nodes at the edge of the network (near to the user). Less popular content must be cached on core nodes of the network. If more popular content will be cached on the nodes near to the user then the content is delivered efficiently in less amount of time as detailed in Fig. 6.1(d). In this algorithm, all the nodes are involved in caching strategy. In other words, some nodes caches popular contents and some nodes cache less popular. They argue that caching contents on all nodes is not a good strategy because it is a wastage of resources.

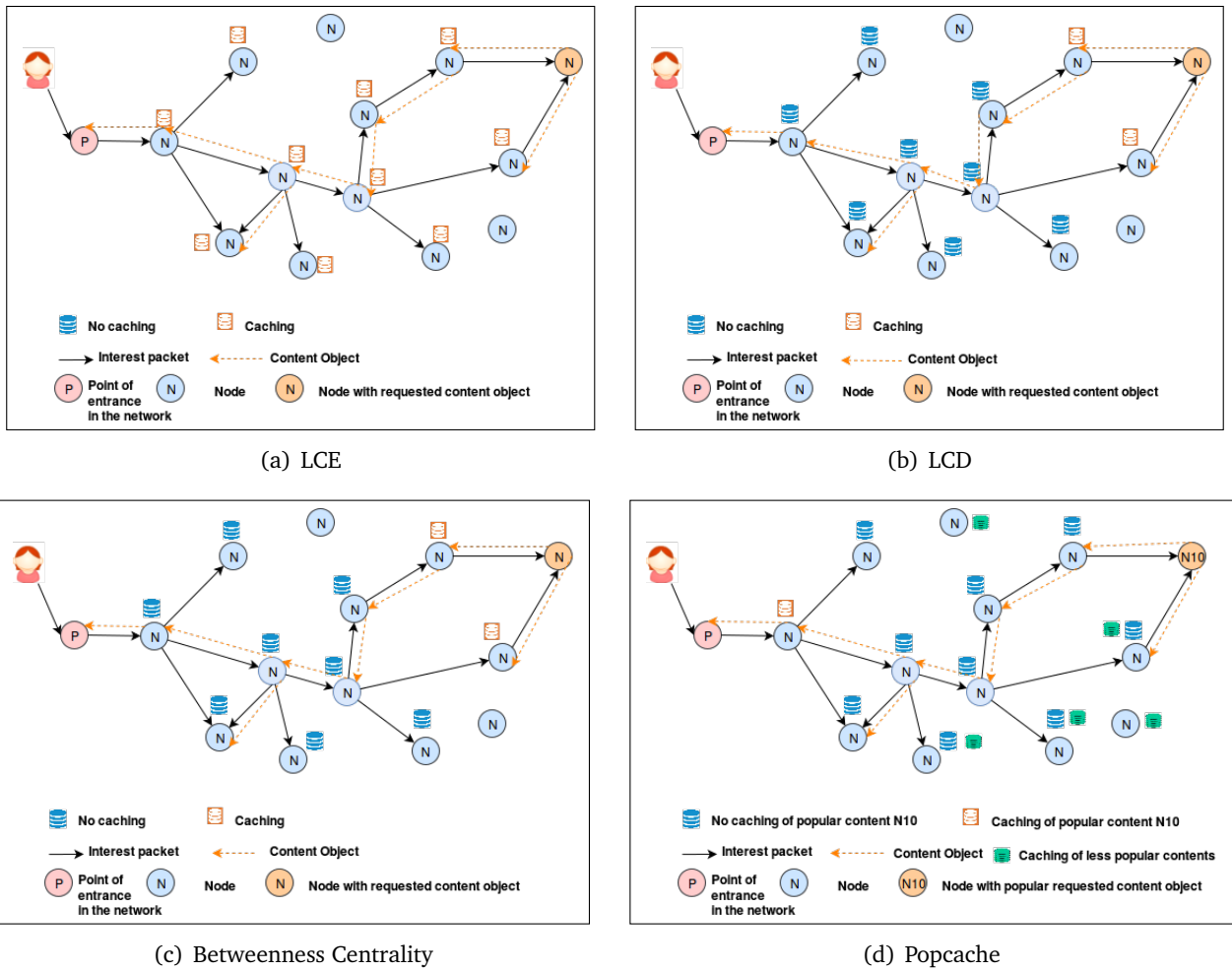


Figure 6.1. Caching placement policies.

6.2.2.3 Cache replacement policies

The design of cache replacement algorithms is realized for content distribution purposes. When the network becomes stable and the node cache overflows, a replacement policy, such as Least Recently Used Least Recently Used (LRU), Least Frequently Used (LFU), First In First Out (FIFO), or Random policy is used to evict one of the cached contents to make room for the newly arrived one.

Least Frequently Used (LFU) when the probability of each data being requested is stationary over time and independent of previous requests, the optimal replacement policy is the Least Frequently Used (LFU).

LFU statically places in the cache the C most frequently requested data [141]. LFU implementation requires content popularity ranking to be known beforehand. However, even without this information, it can still be implemented by keeping counters for each content requested to learn their request frequency as shown in Fig. 6.2. Too many variants of the strategy cited above exist such as Perfect-LFU, in-Cache-LFU, etc.

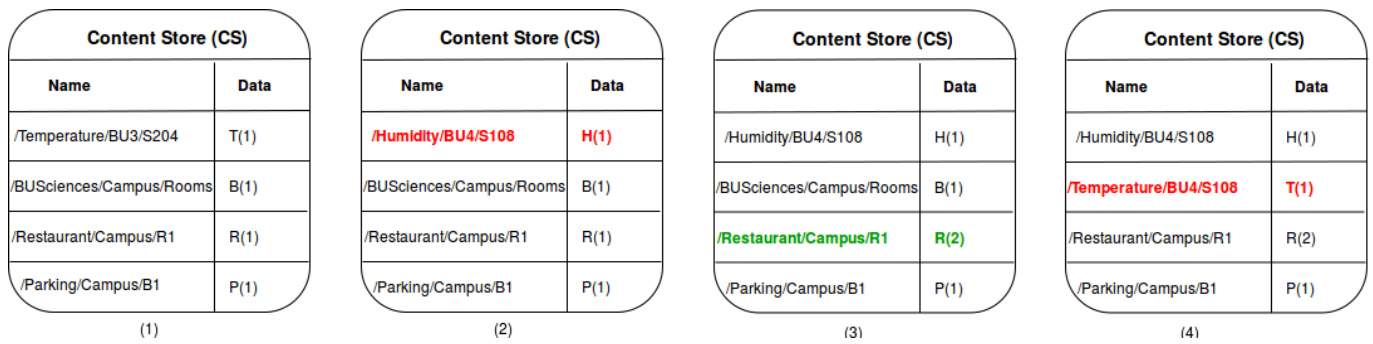


Figure 6.2. Least Frequently Used replacement policy: once the cache is full LFU replaces the least frequently used (frequencies calculated by bits in the data) by most frequented data.

Least Recently Used (LRU) as depicted in Fig. 6.3 in LRU, the idea is to keep the data recently used and to replace with the other data. LRU has two advantages that make it very popular, it is very responsive to non-stationary trends since its replacement decisions are exclusively based on recency and it cannot perform significantly worse than LFU because the ratio between the optimal cache hit ratio and LRU cache hit ratio is bounded [141].

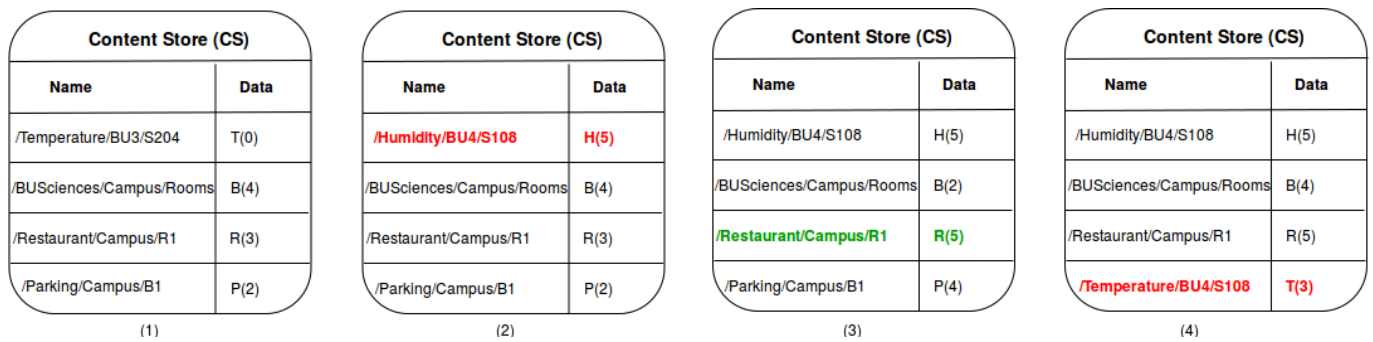


Figure 6.3. Least Recently Used replacement policy: once the cache is full LRU replaces the least recently used (identified by bits in the data) by most used data.

First In First Out (FIFO) when a new content is inserted, the evicted content is the one which was inserted first in the cache as detailed in Fig. 6.4. The behavior of this policy differs from LRU only when a data already present in the cache is requested.

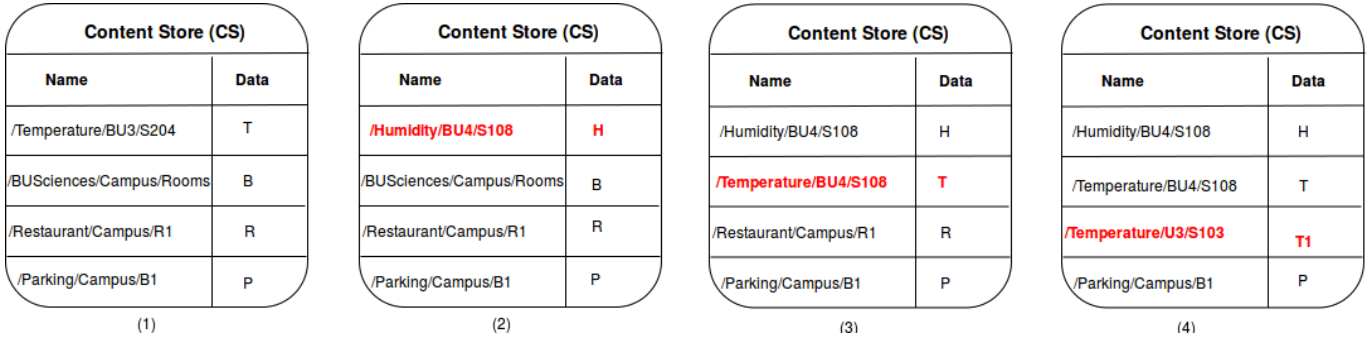


Figure 6.4. FIFO replacement policy: First In First Out replacement policy.

Random policy this policy chooses randomly the content to evict from the node cache.

We assume that the content placement and replacement play a significant role in the resulting traffic and energy reduction. Furthermore, the selection of an appropriate node to cache content so that it could be able to serve future requests for a long time is very important. Consequently, addressing the location problem of caches is an important part of the campaign for in-network caching in CCN. We remind that, in this thesis, we consider sensor environments. Hence, we propose a CCN-WSNs context-aware caching strategy and we start by detailing the model and giving the assumptions of our proposal in the next section.

6.3 A collaborative caching strategy for content-centric enabled wireless sensor networks

6.3.1 Content request process

As already explained in the last chapter, we consider that interests packets follow the Zipf distribution which presents the frequencies of distribution of user interests in the network. This distribution assigns a rank for a popular content. Popularity means that out of all available contents how many times a particular content is accessed. If the content is more popular then its rank is low and if the content is less popular then its

rank will be high. Let $E = N_1, N_2, N_3, \dots, N_{20}$ denote the content population in the system with a size of 20 contents. Since content popularity follows the Zipf distribution, the i^{th} most popular content is requested with a probability proportional to:

$$p_i = \frac{\beta}{i^\alpha} \tag{6.1}$$

where β is the normalized constant with $\beta = \frac{1}{\sum_{i=1}^{20} \frac{1}{i^\alpha}}$ and α is the Zipf exponent.

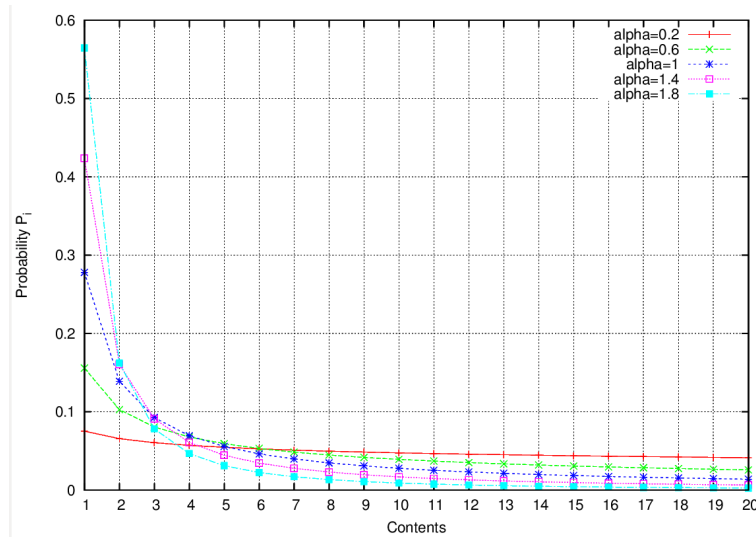


Figure 6.5. Zipf interest distribution depending on the value of α varied from 0.2 to 1.8 for 20 different generated contents.

Fig. 6.5 shows the probability distribution p_i for each content when α varies from 0.2 to 1.8. As shown, when the popularity is low ($\alpha = 0.2$), the probability is nearly the same for all the contents since they have almost the same popularity. Therefore, they behave like a uniform probability law where the probability to request a content is similar. In addition to that, in too many studies, they showed that $\alpha = 1$ refers to a normal popularity where 90% interests request 60% of content. Once the α increases, the probability p_i increases for the most popular contents. For instance, for $\alpha = 1.8$, the content N_1 is requested with a probability equal to 0.58.

6.3.2 Cache admission control

In CCN, in caching strategy area, as already stated, researchers basically try to develop algorithms that

choose appropriate nodes to cache incoming content. CCN supports ubiquitous caching protocol where every node needs to cache incoming content. But sometimes ubiquitous caching is a wastage of resource and it is not a smart strategy to cache the same content on each node. Hence, we need to find a caching strategy to appropriately select node to cache newly arrived content.

When a sensor node receives the requested data or a data goes through it, a cache admission control is triggered to decide whether it should be stored into the cache of the node or not [140]. Inserting a data into cache might not always be favorable because the incorrect decision can lower the probability of cache hit and also makes poor utilization of the limited storage.

We first explored the Steiner Point [146] to find the best location to cache the content. For instance, in [146], they tried to minimize communication cost by finding the nodes of a weighted Minimum Steiner tree. In other words, they created a Steiner Data Caching Tree. They showed that the degree of the node where to cache is 3. Furthermore, they consider that the formation of Steiner data caching tree is done by considering the refresh rates in each edge of the tree. However, they addressed the scenario where multiple subscribers were receiving data from one source. Which is not our case, since users may receive data from multiple nodes. Besides, in the considered scenario, the contents are pushed in the network once they are requested by users. Thus, we do not consider the concept of refresh rate (in this work, an on-demand scenario are considered).

As for [147], the authors contend that node degree is not an interesting insight to consider when caching because in a network of caches the consumer is interested in connecting to the content, not to a specific node. However, we argue that in a wireless network, a node with a high degree may guarantee content availability when it detains a high number of neighbors.

6.3.2.1 Proposed cache placement approach

For the reason cited above, we chose to stay more general and to consider the degree of the node. Hence, in the proposed strategy, the cache admission decision at a node is based on two conditions: (i) the percentage of the path from the source is it greater than Δ ? (ii) if yes, the node does it have a number of neighbors greater than x ? The more the node has an important number of neighbors, the more nodes cache the content and ensure content availability in the network.

Therefore our strategy is a ‘collaborative Caching Strategy Distance and node Degree aware in content-centric enabled wireless sensor networks’ called CSDD.

A trade-off exists between query latency and content accessibility. With a small Δ , the number of copies for each content is high and access delay for this content object is low. On the other hand, with a larger Δ , each content has a small number of copies in intermediate nodes, and the access delay can be longer [140]. However, this depends on the position of the user.

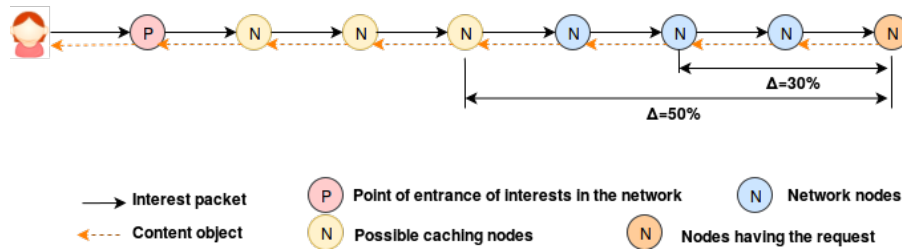


Figure 6.6. The first condition of our caching scheme: distance from the source node.

Fig. 6.6 explains the first condition of the caching strategy and Δ variation; from how much Δ from the source, our strategy decides to cache? In the next section, we will investigate the impact of this parameter in the proposed caching strategy.

Once the first condition is fulfilled, our strategy checks the degree of the nodes on the path, if it is greater than x , the content is cached. Otherwise, the content is forwarded. If the second condition was not verified all the way, the content is cached in the edge node (near the user). For this, a small field is added to the content to check if it was cached on the path or not. When the content reaches the edge node, it checks if it was cached. If yes and if the edge node does not verify the second condition of the strategy, it is just forwarded. Differently, it is cached on the edge node without taking into consideration the verification of the second condition.

Fig. 6.7 shows how the strategy works if a user sends an interest for a content and if Δ is equal to 50% and $x \geq 3$. When the requested content object is found, it is sent back to the user. Meanwhile, it is cached in nodes that fulfill the two conditions related to the degree x and position from the source Δ . As already explained, if the second condition was not verified, the content would be cached in the first node near the user.

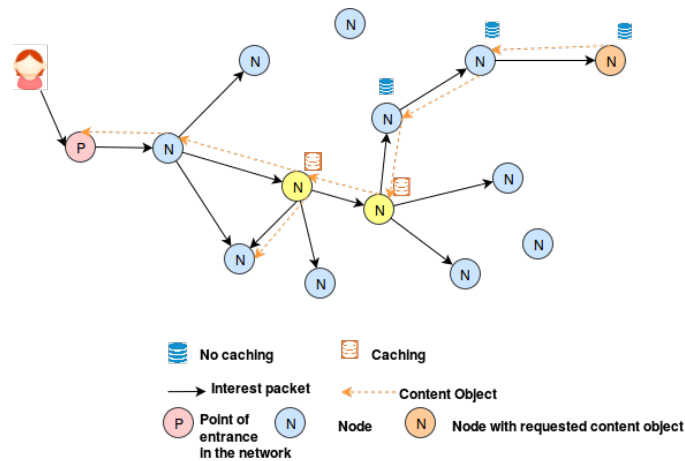


Figure 6.7. An example of function of our strategy when Δ is equal to 50% and $x > 3$.

Loop avoidance

When implementing the caching strategy, we noticed that when replying by a content, since we are working on a wireless communication mode, some contents may be sent two times to a node. In a normal case, when a node receives a content CO , if it does not have CO in its CS, it caches CO , otherwise, it forwards CO . Since in our strategy, the content is not cached in all the nodes of the network. If a node does not cache the received content (because it does not verify the condition of the strategy), it just forwards it. The next nodes who receive the interest will also forward the interest (whether cached or not). Consequently, the nodes that already have received the content but did not cache it, when receiving an interest for the same content, repeat the same process. Which causes a problem of loop hence a waste of energy consumption. To resolve this problem, we proposed to give an ID to every generated interest and this to avoid treating the same interest for a defined time.

6.3.3 Cache replacement policy

Our mechanism uses a weak consistency model ages travel by multiple hops, it is important to ensure high reliability. Once the CS is full, our replacement policy relies on replacing the less popular content in the node content store with the new content in the content store. It aims to keep popular contents in the CS. Then our replacement policy is Popularity-based. In addition to that, the interests follow the Zipf distribution.

In our replacement policy, we suppose that popularity denoted by P of content $N1$ is bigger than the

Algorithm 3: Proposed caching strategy GSDD

```

input   :  $\Delta$ : Distance from the node having the content
            $n$ : Number of neighbors
            $x$ : Threshold set for the number of neighbors

//A user send an interest  $I$  for a content  $co_i$ . Once the interest is received on a node, a
//counter is set. When the interest is forwarded from a node to another the counter is
//increased by 1.
1: Receive an Interest  $I$ ;
2: Set counter  $t = 0$ ;
3: while  $I$  is forwarded from a node to another do
    //t will help in calculating the distance crossed by the interest to attend the node
    //having the corresponding content.
4:      $t++$ ;
5:     if the content is found in the CS then
        //t corresponds here to 100% of the path.
6:         Assign  $t$  to the content;
        //d is the required distance (from the node having the content) from where our
        //strategy decides to start caching.
7:         Calculate  $d = \frac{t*\Delta}{100}$ ;
8:         while the content is forwarded from a node to another do
9:              $t--$ ;
            //Check the first condition of our protocol concerning the distance from the
            //node having the content.
10:            if ( $t \leq d$ ) then
                //Check the second condition of our protocol concerning the degree of the
                //node.
11:                if ( $n \geq x$ ) then
12:                    Cache the content;
13:                    Mark the content as already cached;
14:                     $f = TRUE$ ;
15:                if ( $t = 1$ ) then
16:                    if ( $f = FALSE$ ) then
17:                        Cache the content on edge node;

```

popularity of N_2 . Then, $P(N_1) > P(N_2) > P(N_3) > \dots > P(N_n)$. Since the cache size is limited, the node only caches the most popular content and evicts the less popular. If two contents of the same popularity

exist when applying replacement, our strategy evicts the content having the smallest index for example if $N8$ and $N7$ have the same popularity $N7$ is evicted.

6.4 Performance evaluation

In this section, we examine the performance of CSDD under different degree values x and distance Δ . We then compare it to LCE and LCD, under two replacement policies FIFO and Popularity-based. Moreover, we varied the Zipf exponent α . Other factors are to study such as the network size and the cache size but we made the choice to start with these based on the results that we published in an internal research report [148].

6.4.1 Simulation set-up

For the implementation of our the existing strategies and the proposed scheme, we continue on working with the framework CCNx_Contiki [70] and we modify it to follow the requirements of all the strategies. In the simulation, too many users poll their interest through 4 points of entrances in the network.

| Simulation Parameters | Value |
|--|------------------------|
| Area | 500 $m \times$ 500 m |
| Simulation duration | 3600 s |
| Radio coverage range | 100 m |
| Initial energy | 2 J |
| Cache size | 6 (30%) |
| Number of generated types of content | 20 |
| Number of entrance points in the network | 4 |
| Values of α | 0.2, 0.6, 1, 1.4, 1.8 |

Table 6.1. Simulation parameters

Sensor nodes are static and the communication links are bidirectional. For sake of simplification, we suppose that the contents have the same size. As already mentioned, the packets follow the Zipf distribution [129].

Since our strategy is based on the node degree, we study the percentage of nodes that have a certain number

of neighbors and we plot the results in Fig. 6.8. Therefore, more than 57% of the node network present a degree higher than 2 and 16% of nodes detains a degree higher than 3.

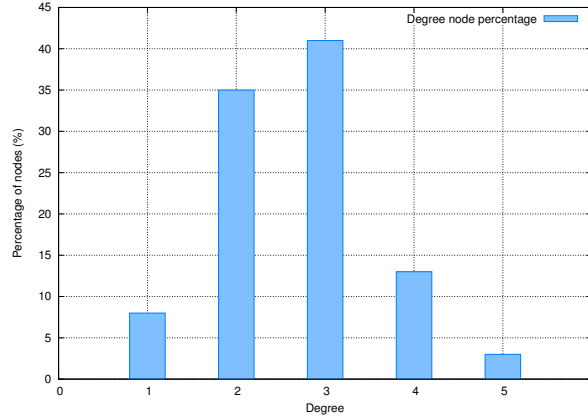


Figure 6.8. Percentage of nodes detaining a certain node degree.

6.4.2 Evaluation metrics

We have previously introduced the energy consumption which presents an important metric in our study. Consequently, in this chapter, in order to evaluate the proposed strategy, we continue on evaluating the energy consumption under the model proposed in the chapter 5, section 5.4.2. We then consider other new metrics such as the network lifetime, the stretch, the cache diversity, and the cache replacement rate.

The notion of network lifetime chose in this study presents the duration until the first node exhausts all its energy [149].

We also present results for the Stretch which defines the percentage of the path that has been crossed to retrieve the content [150].

$$Stretch = \frac{\sum_{i=1}^I hops_crossed_i}{\sum_{i=1}^I total_hops_i} \quad (6.2)$$

Where I defines the number of the total generated contents.

The next metric is the cache Diversity measures the number of distinct elements stored in the caches. It expresses the ratio between the cardinality of unique contents stored in all caches and the cardinality

total number of contents in the caches [150].

$$Diversity = \frac{Card \bigcup_{n=1}^N CO_n}{\sum_{n=1}^N Card CO_n} \quad (6.3)$$

Where N defines the total number of nodes in the network and CO is the content object.

Besides, we measure the cache replacement rate that presents the ratio between the replaced contents and the cached contents for all the nodes in the network.

$$Rep_rate = \frac{\sum_{c=1}^C replaced_contents_c}{\sum_{c=1}^C cached_contents_c} \quad (6.4)$$

With C the total number of contents requested.

Finally, we measure the cache hit measured on a path when looking for a content [150]:

$$Cache\ hit = \frac{\sum_{i=1}^N hits_i}{\sum_{i=1}^N hits_i + \sum_{i=1}^N miss_i} \quad (6.5)$$

With N the total number of nodes in the network.

6.4.3 Simulation results

In this study, we implement two cache replacement policies: the FIFO and the Popularity-based strategies. Since our interests follow Zipf distribution, we consider that different α describes different scenarios.

$\alpha \leq 0.6$ describes the low popularity scenarios when there is no rush like students asking for the temperature of a classroom or for information about teachers. This can happen all along the day and by a small number of users at the same time. However, $\alpha \geq 0.6$ presents high popularities scenarios describing rush hours such as 11.30 *am* when many students start sending interests to have an idea about the menu and the queue in front of the restaurant.

Energy consumption

Figure 6.9 plots the results of energy consumption for the different strategies under different variation of α for two replacement policies: FIFO and Popularity-based.

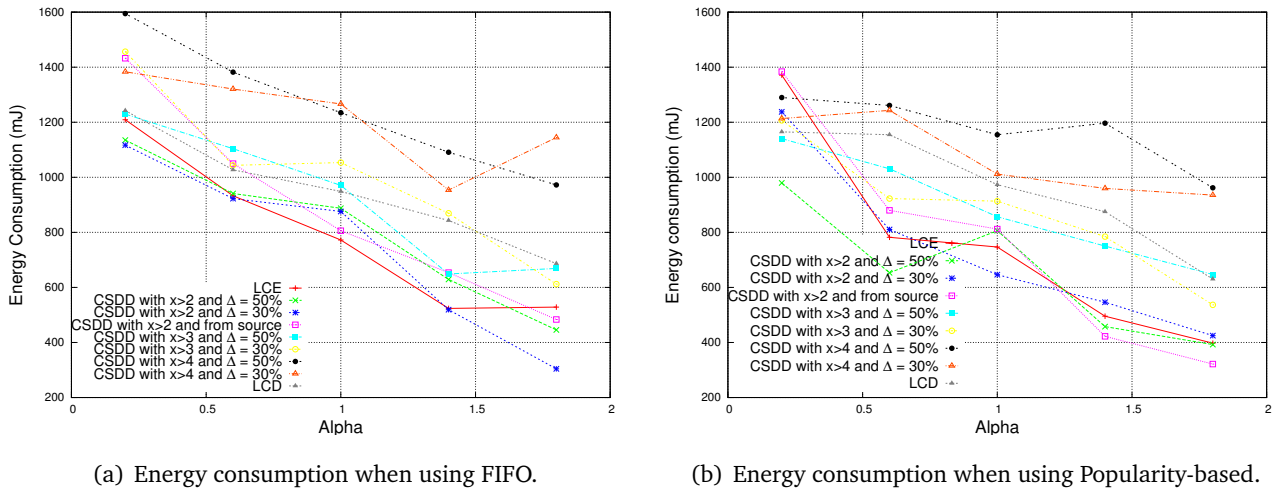


Figure 6.9. Energy consumption for all the strategies when using FIFO and Popularity-based replacement policies for 80 nodes.

For both replacement policies, it is noticed that when α increases the energy consumption decreases for all the strategies under both replacement policies. Indeed, when α increases, some contents become more popular than others. Hence, they will be more requested by users and they will be cached more in intermediate nodes. Consequently, most of the requests cross a shorter path. They are not required to reach source nodes to get the corresponding contents. Then, energy consumption decreases. In low popularity scenarios, all the strategies consume more energy. This is due to the fact that the contents are almost requested with the same rate. Then, the replacement happens frequently increasing energy consumption. It is also worth to note that under the FIFO replacement policy, all the strategies consume more energy. This is explained by the fact that since the interests follow Zipf distribution in both cases when using FIFO, the probability of replacement is the same for all the contents. Nevertheless, when using the Popularity-based policy, popular contents will be kept in the caches. Therefore, the network consumes less energy consumption since the requested contents will be in the cache for a longer time. For instance, for $\alpha = 0.4$,

under FIFO, LCE consumes about 1230 mJ and CSDD ($x > 2$ and $\Delta = 30\%$) consumes slightly more than 950 mJ . Yet, under Popularity-based policy, CSDD ($x > 2$ and $\Delta = 30\%$) dissipates just 890 mJ and LCE consumes 1050 mJ .

For low popularities, CSDD with a node degree higher than 2 and a caching distance from 30% or 50% consumes less energy than LCE, LCD, and CSDD with a degree of 3 or 4. Since nodes with a degree higher than 3 represent only 16% of the network nodes, contents will not be cached on a lot of nodes then, requests will be forwarded to many nodes to find the content. Then, CSDD with a degree of 3 or 4 consumes more energy.

For FIFO policy, when the popularity of contents is high, LCE acts better than all the other strategies in terms of energy consumption since it caches popular contents everywhere. However, when the Popularity-based policy is used, for high α , CSDD with a x (node degree) higher than 2 and a Δ equals to 50% outperforms LCE. Concerning LCD, we observe that it consumes more energy than LCE and CSDD with $x > 2$ and $\Delta = 30\%$ or 50%. Indeed, LCD just caches contents one hop from the source. Then, when an interest arrives on the network, it has to go to one hop from the source to get the corresponding content.

It is also interesting to mention that in our strategies, only nodes with a degree $x \geq 2$ or $x \geq 3$ caches contents. Hence, the number of nodes that caches the contents and communication computation decreased. Then, caching energy is saved. However, sometimes more forwarding energy is dissipated when the subset size of the nodes having this degree is not significant.

Stretch

Fig 6.10 represents the results for stretch for LCE, LCD, for our strategy under different variation of Δ and x .

As shown, when FIFO policy is used, in low popularity context, CSDD with $x > 2$ and a $\Delta = 30\%$ or $\Delta = 50\%$ outperforms all the other strategies. This is due to the fact, that the content is cached in nodes near to the user and the number of nodes having a degree larger than 2 represents more than 60% of network nodes. Then, crossed distance become smaller and probability to cache in nodes with a $x > 2$ is large.

The results are quite the same for the Popularity-based replacement policy. Yet, CSDD with $x > 3$ and a $\Delta = 50\%$ outperforms all the strategies but there is no a big difference in the values of the stretch.

Findings for LCD shows that it has the biggest stretch for both policies. Indeed, since it caches one hop from the source node, every time, the interests have to cross almost all the path to recover the content.

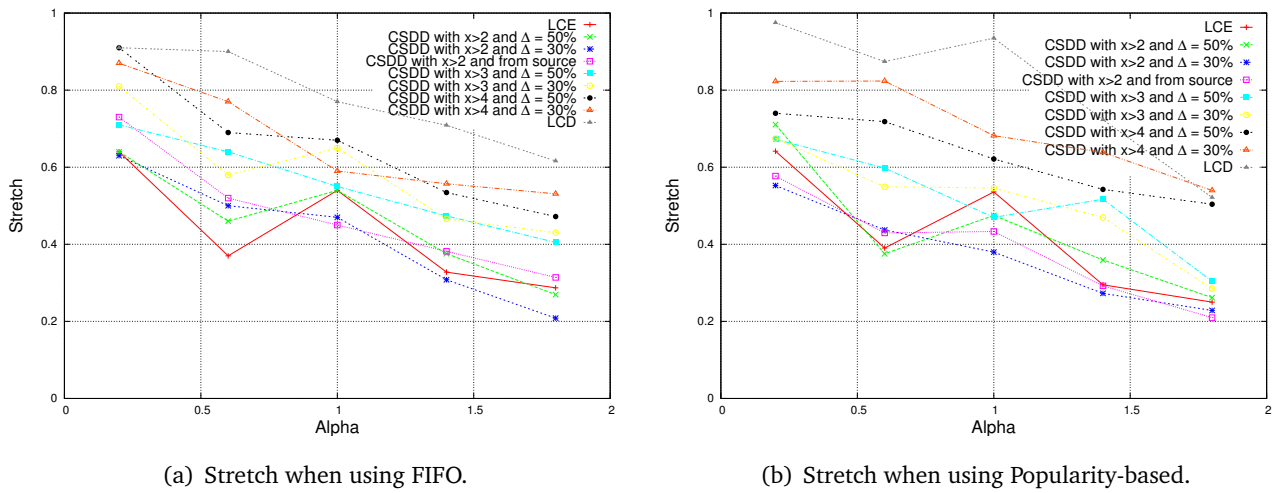


Figure 6.10. Stretch for all the strategies when using LCE and Popularity-based replacement policies.

Furthermore, when the popularity of certain contents increases (α increases), the stretch decreases. Certainly, this happens because popular contents will be more available in intermediate nodes.

Network lifetime

FIFO replacement policy: We illustrate in Fig. 6.11 the network lifetime for different α under FIFO replacement policy.

As noticed, LCD presents the worst results since it always caches in a few particular nodes. Therefore, the node battery expires yet.

In the context of low popularity ($\alpha < 0.6$), CSDD with $x > 2$ and a $\Delta = 30\%$ or $\Delta = 50\%$ and even with $x > 3$ outperforms all the other strategies even LCE. This is explained by the fact that in CSDD, fewer nodes than in LCE caches the content, all contents have almost the same popularity and the cache size is limited to 6 for all the strategies. Therefore, LCE which caches everywhere realizes replacement frequently and exhausts quickly node battery. For instance, for LCE, the first node die at $t = 1400$ s.

In contrast, in the case of high popularity as depicted in Fig. 6.11(e), the performance of LCE becomes better but it still does not overcome CSDD with $x > 2$ and a $\Delta = 30\%$ or $\Delta = 50\%$. The battery expiration of the first node happens at $t = 1600$ s in this case. Then, when the popularity increases, the network lifetime increase since popular contents will be more available in intermediate nodes when they are requested.

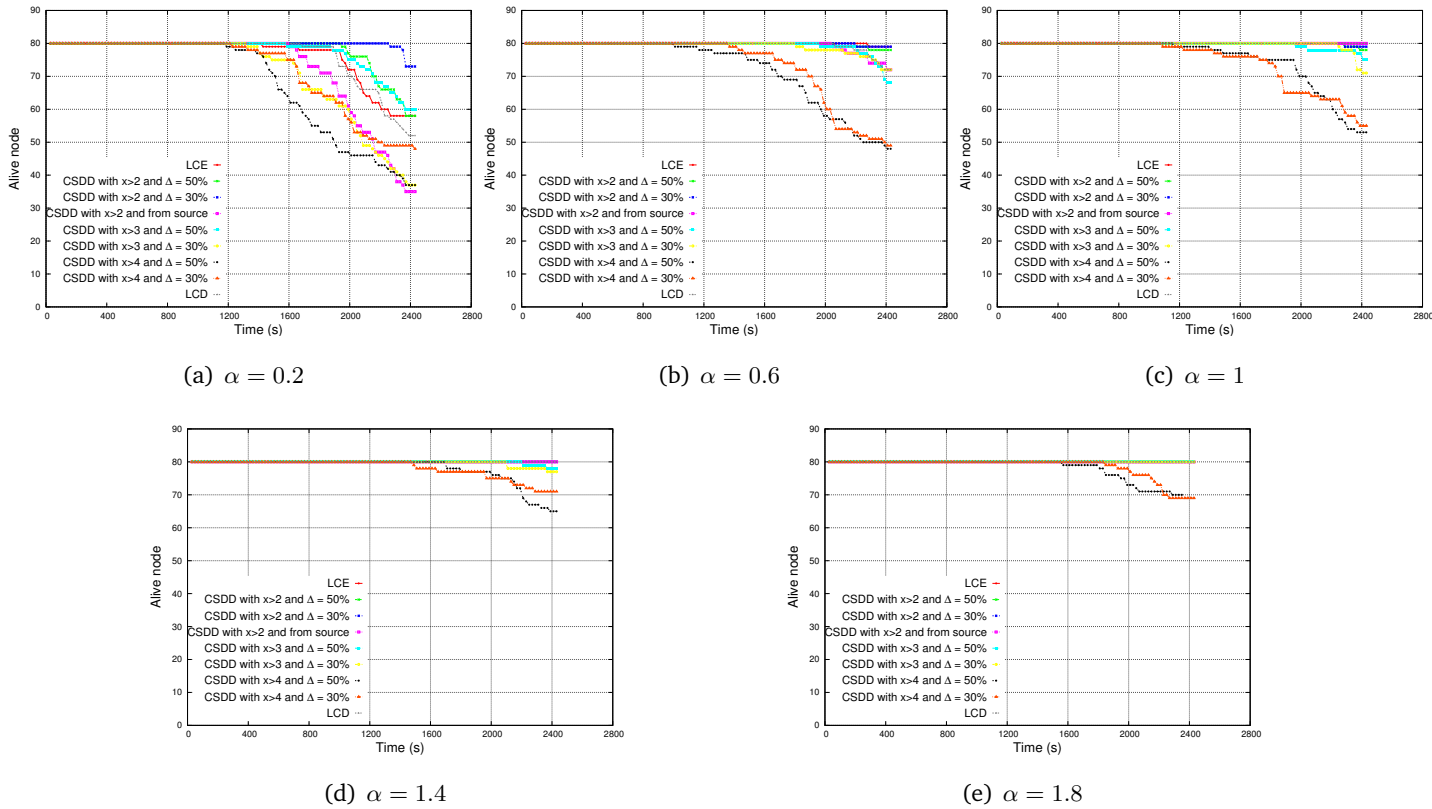


Figure 6.11. Network lifetime for different α for all the strategies when using a FIFO replacement policy.

Concerning CSDD with $x > 3$ and $x > 4$, they have the worst network lifetime, since the number of nodes having these criteria is not big (it just presents 16% of the node network), then caching will always happen on these nodes causing their exhaustion early.

Popularity-based replacement policy: Fig. 6.12 details the findings for network lifetime for different α when using Popularity-based replacement policy.

As depicted in the figures, the network lifetime is enhanced when applying this type of policy. Comparing to the results shown in the previous Fig. 6.11(a), the proposed policy enhances the results when the popularity is high.

In the other hand, in the context of low popularity, CSDD with $x > 2$ and a $\Delta = 30\%$ or $\Delta = 50\%$ still overcomes all the other strategies even LCE. But, the results are quite the same, for example for $\alpha = 0.2$,

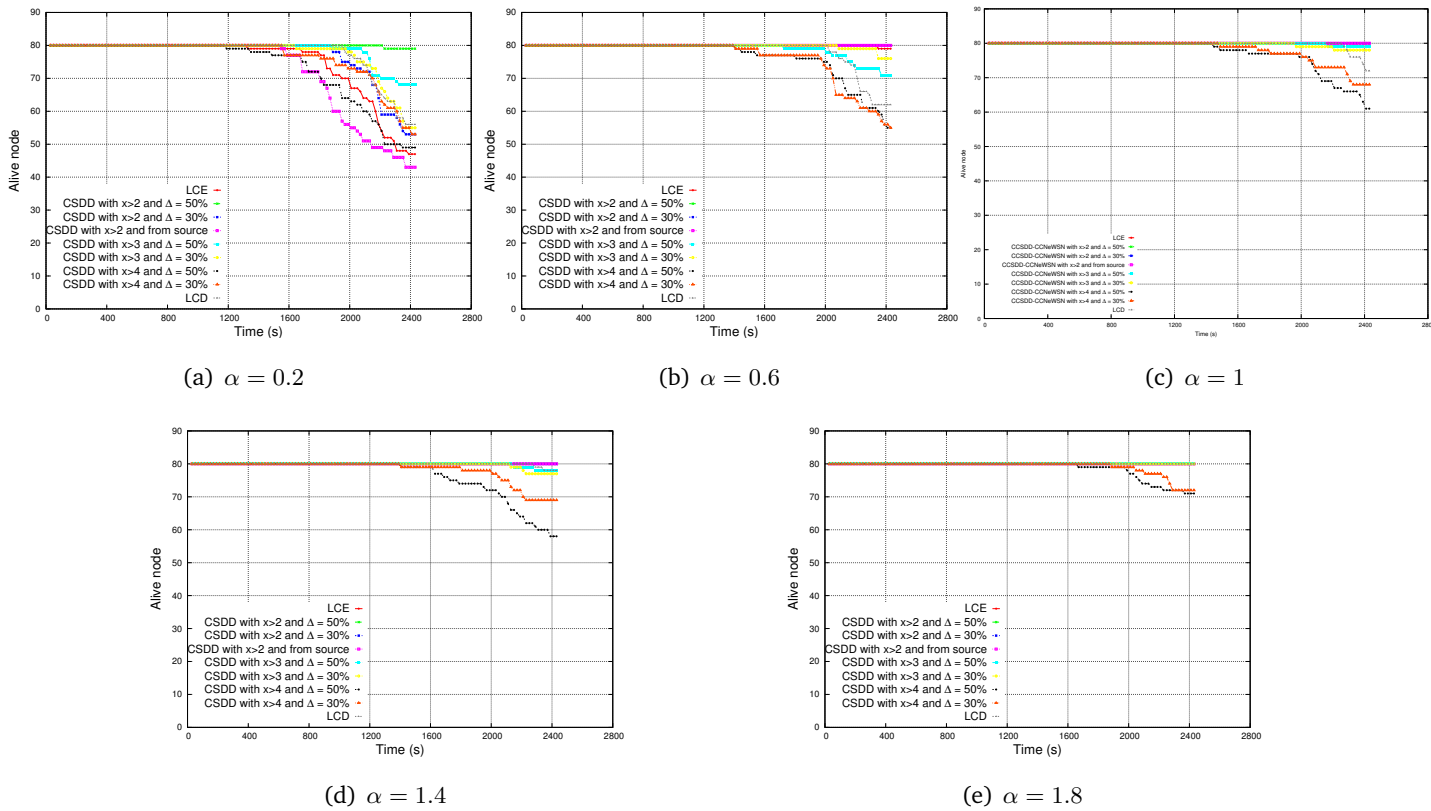


Figure 6.12. Network lifetime for different α for all the strategies when using a Popularity-based replacement policy.

the first node dies in LCE at $t = 1450$ s. Because with low popularity scenarios, contents almost have the same popularity and the replacement rate does not impact a lot. However, when the content popularity increases, LCE starts acting like the proposed strategy (with $x > 2$ and a $\Delta = 30\%$ or $\Delta = 50\%$) and we do not notice any node exhaustion during the duration of the simulation. This is because interests follow the Zipf distribution and the applied replacement policy is Popularity-based. Consequently, replacement happens less frequently and even so interests do not cross the whole path till the source. Then, node energy is saved since the content is available in intermediate nodes.

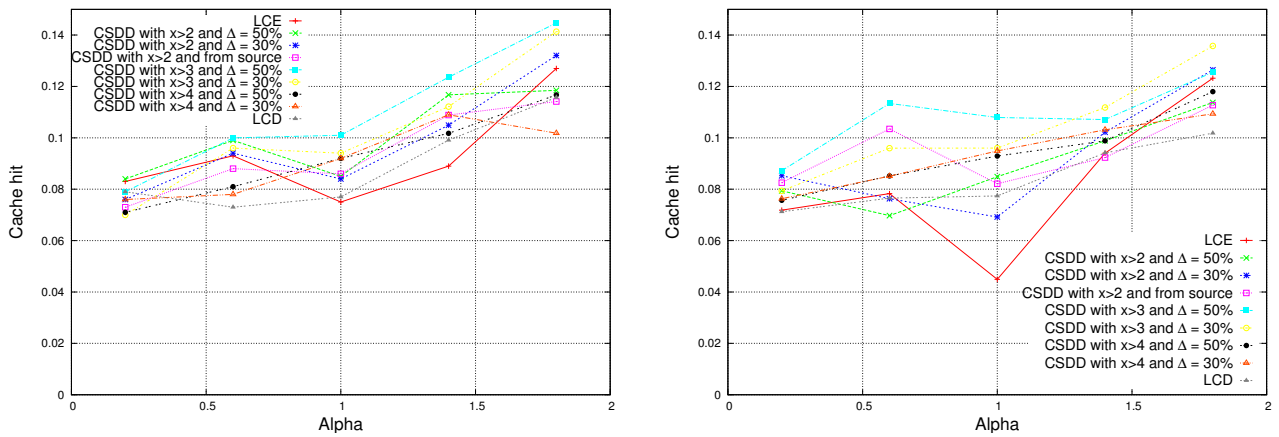
It is also worth to note that when nodes start dying, the energy consumption increases, since off nodes will not be able to ensure their functionalities, then the interests cross longer paths to recover the contents.

The LCD network lifetime is enhanced when using the Popularity-based but it is still not better than LCE

network lifetime.

Cache hit ratio

We also studied the cache hit ratio for all the strategies under different popularity distribution when using both FIFO and Popularity-based replacement policies.



(a) Cache hit when using FIFO.

(b) Cache hit when using Popularity-based.

Figure 6.13. Cache hit for all the strategies when using FIFO and Popularity-based replacement policies.

The results plotted in Fig 6.13 have shown that when the popularity increases, the cache hit increases when using FIFO and Popularity-based. Indeed, when the α increases, the popularity of certain contents increases and they will be more requested hence they will be more available on the network node.

In Fig. 6.14(a), for $\alpha = 0.2$ when FIFO is used, we observe that LCE outperforms almost all the strategies (except CSDD with $x > 2$ and a $\Delta = 30\%$ or $\Delta = 50\%$). This is due to the uniform distribution of interest and to the maximum caching in nodes. However, in high popularity, CSDD with $x > 3$ and a $\Delta = 30\%$ or $\Delta = 50\%$ reports better performance than LCE.

For Popularity-based replacement policy, for low popularity, this time LCE presents the worst result because it is caching everywhere the same contents. However, in high popularity, it is achieving better results since during the rush hours, there are a lot of interests sent on the network requesting the most popular contents. So, it is better to make them available. However, LCE still does not outperform CSDD with a $x > 2$ or $x > 3$ in this case. Therefore, we notice that the values of the caching are low since, in a wireless network, the

interests are widely diffused increasing the number of miss since the requested content is not cached in all the network node. Then, they will be recording cache miss and decreasing the cache hit ratio.

Replacement rate

In Figure 6.14, findings for the replacement rate ratio are plotted for all the implemented and proposed strategies when using FIFO and Popularity-based replacement policies.

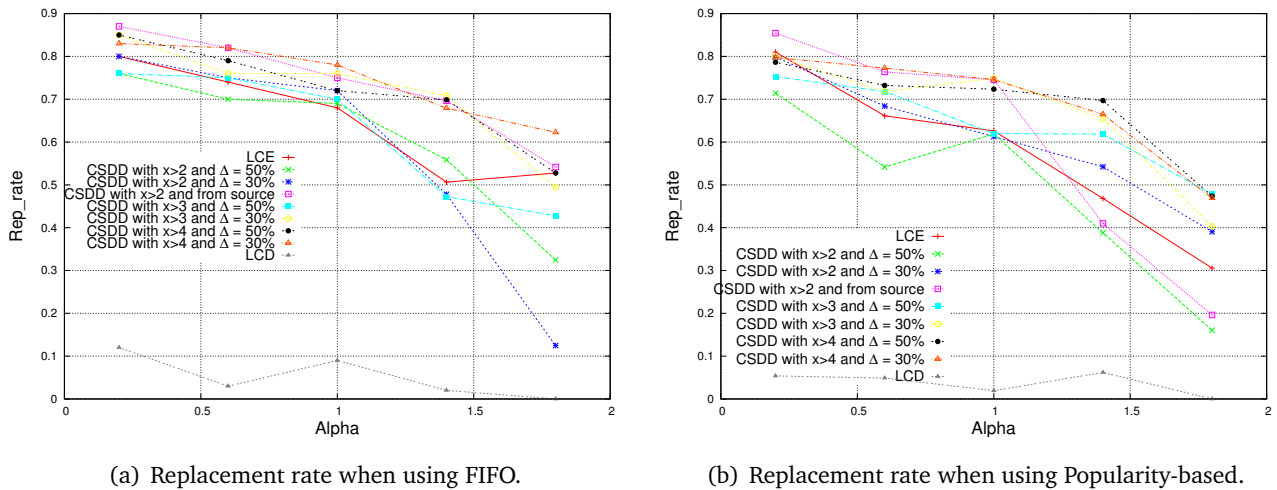


Figure 6.14. Replacement rate for all the strategies when using FIFO and Popularity-based replacement policies.

As shown, when the popularity increases, the replacement rate for both policies decreases. In fact, when α is high, just popular contents are requested. For instance in this simulation, content $N1$ is requested with a probability equal to 0.58 for $\alpha = 1.8$. Then, content $N1$ is available in intermediate nodes and its replacement rate is small.

For FIFO policy, when the popularity is low, CSDD with $x > 2$ and a $\Delta = 30\%$ or $\Delta = 50\%$ realizes better results than LCE. However, in case of high popularity, LCE becomes better because popular contents will be cached everywhere and replacement operation will not be frequent. As depicted in the figure, the values of the replacement rate are better under the Popularity-based replacement policy because of its capacity to replace with the less popular content.

It is noted also that LCD, in case of low or high popularity for both FIFO and Popularity-based replacement policies, outperforms all the implemented strategies even LCE. As mentioned before, LCD caches only in

one hop from source nodes and then replacement only occurs in these nodes. Then, the overall replacement rate is low.

Diversity

Finally, we investigate the diversity for different $\alpha \in \{0.2, 0.6, 1, 1.4, 1.8\}$ for all the strategies when using FIFO and Popularity-based replacement policies. Indeed, strategies that ensure high diversity, may satisfy interests requesting contents with low popularity.

FIFO replacement policy: Results for FIFO are plotted in Fig 6.15.

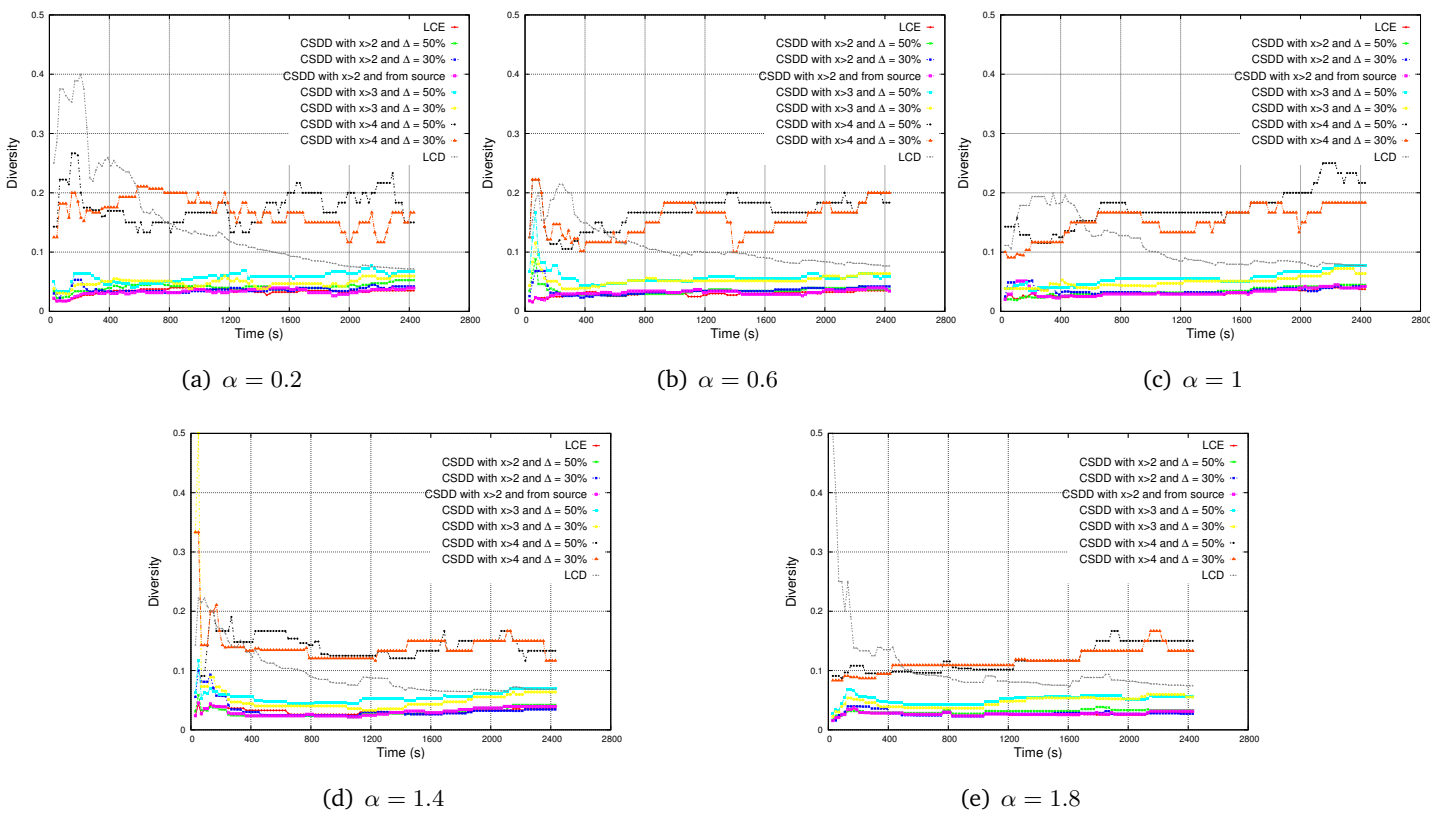


Figure 6.15. Diversity for different α for all the strategies when using a FIFO replacement policy.

It is observed that when α increases, the cache diversity decreases. Indeed, in case of high popularity, as mentioned earlier, specific content is more requested and cached. Besides, FIFO ejects from cache contents

with the same probability decreasing also the diversity.

In the case of low popularity, we note that LCD present better results than the other strategies at the beginning of the simulation since it caches only on particular nodes which are one hop from the user. At $t = 1200\text{ ms}$, CSDD with $x > 4$ and a $\Delta = 30\%$ or $\Delta = 50\%$ shows better results until the end of the simulation with a diversity larger than 0.2. Absolutely, this is because this strategy hides on a set of very limited nodes. Then, it caches less than the other strategies which increase the diversity. In high popularity, as depicted in the Fig .6.15, the diversity slightly decreases since interest distribution (asking for popular content) provides only popular content in caches.

Popularity-based replacement policy: Results for Popularity-based are shown in Fig 6.16.

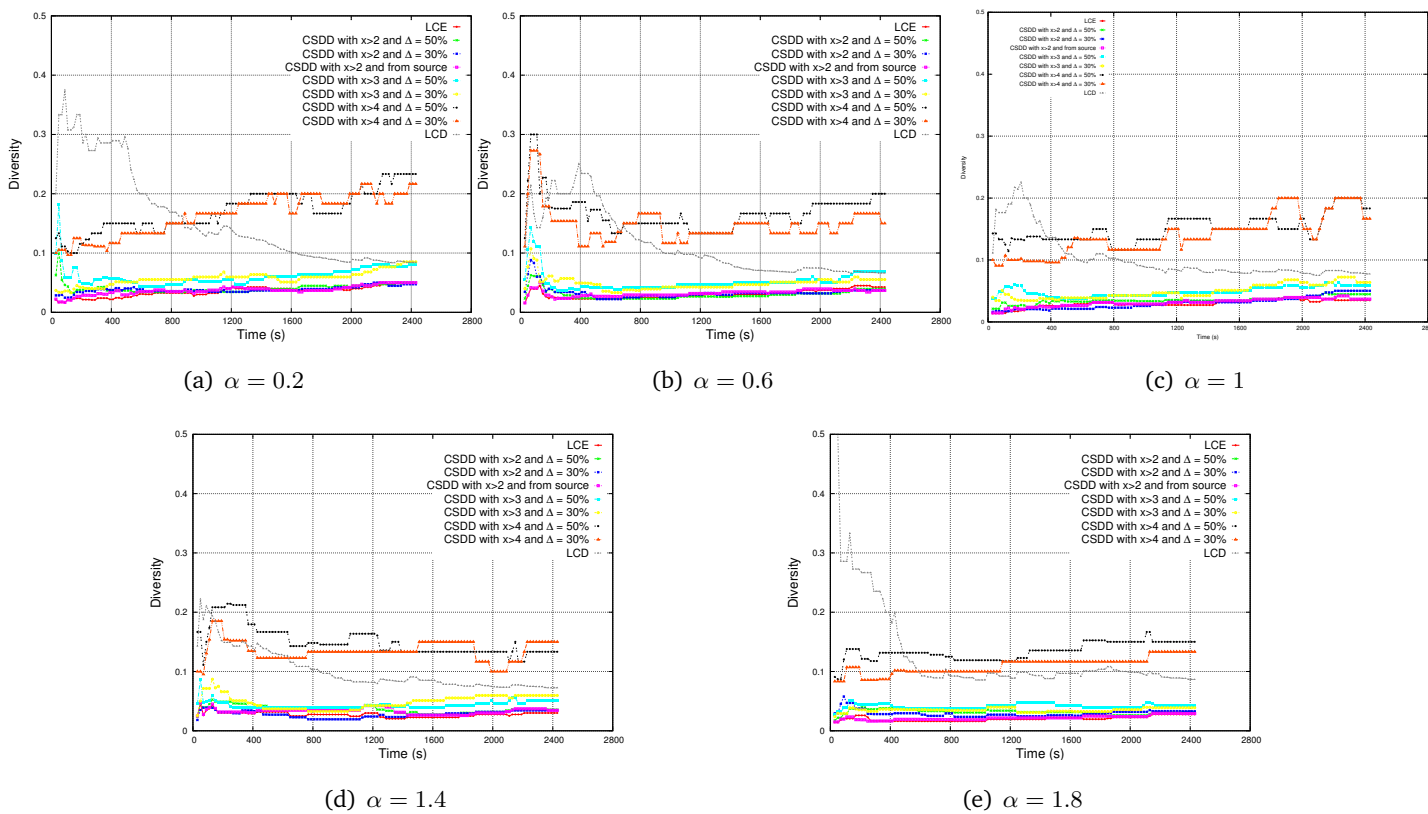


Figure 6.16. Diversity for different α for all the strategies when using a Popularity-based replacement policy.

An interesting observation comes from Fig 6.16, the application of the Popularity-based replacement

policy decreases the cache diversity. Indeed, this replacement policy enables the caching of the same popular contents. The difference is slightly observed in the case of low popularity but it is obvious in high popularity. For both content popularity scenarios, LCE shows the worst results in term of cache diversity. The diversity expresses the ratio between the number of unique contents stored in all caches and the total number of contents in the caches. Then, when the number of contents in the caches increases, the diversity decreases. This explains the results achieved by LCE. This also explains why CSDD under different x and Δ realizes better results.

For high popularity, LCD outperforms all the strategies and achieves a diversity of 0.5. This is because it caches just one hop from the source node the content which decreases the total of cached contents. Besides, since we have different users asking for different contents, diversity is ensured all over the network.

6.4.4 Discussion

The simulation results showcased the results of CSDD under variation of several parameters (replacement policy and α) and compared to LCE and LCD. The proposed caching strategy is based on two parameters, x the node degree and Δ its distance from the source.

For energy consumption, the results highlighted that CSDD with $x > 2$ and $\Delta = 30\%$ or $\Delta = 50\%$ achieves energy saving under the two replacement policies compared to the other strategies. This is due to the fact that the percentage of nodes having this degree is 60% and minimizing caching saves energy. However, CSDD with a $x > 4$ presents the worst results since the percentage of these nodes is just 4% then the content is not cached a lot on the network which increases the energy used while forwarding the data. As for the network lifetime, CSDD with $x > 2$ and $\Delta = 30\%$ or $\Delta = 50\%$ shows the best results for both replacement policies and under different values of α . Although LCE shows good results in term of energy, it realizes bad network lifetime since nodes start dying early. Yet, for a high α , it seems to have better results.

The results have also shown that CSDD can compete LCE in terms of cache hit. In other words, while using the Popularity-based replacement policy and under low popularities, CSDD with different x and Δ shows a better cache hit ratio. In high popularities, CSDD with $x > 2$ and $\Delta = 30$ or $\Delta = 50$ outperforms LCE.

As for the replacement rate, CSDD with $x > 2$ and $\Delta = 30$ or $\Delta = 50$ and LCD outperforms LCE. This lay the groundwork for caching in all the node in LCE and then realizing replacement in all the nodes path

if caches are full.

Our analysis for the diversity also showed that by applying CSDD, better diversity is achieved. In high popularity scenarios, the diversity decreases since just popular contents are requested. When FIFO is used, results for diversity are better because the Popularity-based evicts the less popular contents.

Finally, we argue that the choice of the degree must be coherent with the percentage of nodes having this degree. For instance, the results showed that the degree $x > 4$ realized bad results since it just represents 4% of the network nodes. Then, the number of potential candidates on which caching is realized, is low. We also contend that our strategy can be enhanced by considering user mobility. In fact, in CSDD, the mobility will generate a change in the nodes on which to cache which will maximize the network lifetime. The replacement rate will also decrease and the cache diversity will increase.

6.5 Conclusion

Cooperative caching can play a major role in handling effectively the queries and in overcoming the situations of none availability of data. Therefore, it reduces the requirement of wireless bandwidth, storage, and energy. In-network caching is one of the best features offered by the content-centric paradigm and one of the most characteristics that motivated us to enable CCN in WSNs. CCN was introduced with LCE which caches contents everywhere realizing a certain degree of redundancy. In this chapter, we proposed a new caching strategy that aims to find an optimal way to cache the content in order to realize better network performance. For this, we started by presenting different existing caching strategy in the literature that we implemented further. After that, we proposed CSDD an on-path caching strategy in content-centric enabled wireless sensor networks with two parameters to vary (node degree and its distance from the source node). We also proposed a mechanism to detect and overcome the data loop problem caused by the broadcast. Finally, we presented the simulation results in order to show the impact of CSDD on energy consumption, cache hit, replacement rate and diversity. We also realized a comparative study with LCE and LCD. The results showed that CSDD can outperform LCE and LCD when the degree x detains a subset of nodes with an important size.

Conclusions and Perspectives

7.1 Conclusion

ICN paradigm is considered to be the architecture of the future Internet of Things because it gives up on the notion of content address and adopt the notion of content name. Hence, data becomes independent from location, application, storage, and means of transportation enabling in-network caching and replication. ICN offers multi-hop forwarding, caching capabilities and content verification integrity. It can be deployed in a different environment and supports a wide range of networked applications and dense networks. Thanks to all these features, the research community found that it could bring a solution for some communication constraints and hardware limitations raised in wireless sensor networks, the IoT most important backbone.

Wireless sensor networks suffer from the constraints set by sensors and the medium which is based on the broadcast. Since WSNs already provides information-centric services, ICN paradigms have attracted the attention of the WSNs community to incite the use of information based concept. Among several proposals, it was shown that content-centric networking could be a good candidate to be integrated into WSNs. This thesis demonstrated that the integration has to follow some essential steps to realize successful adaptation and take benefit of such an approach for these networks. Indeed, CCN offers a structure that could enhance the communication in this type of network as well as a caching potential that could solve several problems posed by the memory limitation of sensors.

However, although the advantages that CCN could bring when enabled in WSNs, in some cases, the growth in the requirements wide variety of applications for wireless sensor networks could impact negatively the growth and the enhancement of content-centric paradigms. Most of WSN applications are implemented in the field of critical data monitoring. Consequently, these type of applications needs fresh data to ensure the reliability of the exchanged contents and to respond to user expectation. Besides, since in several applications, data is updated frequently, data copies in node cache will be stale. This requires a

strong and effective approach to realize content update merged with an efficient caching strategy. Besides, the caching strategy has to ensure content availability whenever requested with a low latency to fit WSNs requirements. In addition to that, WSNs are based on a wireless communication medium that supports one interface so it is difficult to reduce the amount of data that needs to be transmitted to the sink. Therefore, data redundancy and energy waste occur. To overcome this, an adaptive routing and forwarding approach that fits content-centric architecture and WSNs needs have to be found.

7.2 Contributions summary

During this thesis, our goal was to enable a CCN in WSNs to improve data dissemination and to treat the constraints posed by these types of networks. This thesis addressed three major problems, the problem of content freshness in caches, the problem of data forwarding in such networks and how it has to be adopted to meet the requirement of the paradigm and the network at the same time, and the problem of content placement that can be solved by the most important feature offered by CCN which is in-network caching.

The first contribution was related to how to ensure the content freshness in the node content store in a content-centric wireless sensor network. In this context, we proposed DFCCN which integrate the lifetime of each content in the network. After that, based on DFCCN, we proposed two approaches for content update, a reactive RDFCCN, and a proactive one PDFCCN. The reactive approach depends on the traffic and the proactive is traffic independent. Results show that not only our approaches ensures network reliability by ensuring data availability but also improve the system performance in terms of delay.

The second contribution addressed the energy consumption model and the impact of forwarding in CCN enabled WSNs. We started by proposing an energy model that fits with the CCN architecture and content forwarding schemes. We then proposed an adaptive duty-cycling approach called as ADDC-CCWSN. This latter implements a mechanism that decides to reduce the duty-cycle of nodes that do not satisfy user interests depending on a certain threshold. Results under different thresholds showcased that even when the duty-cycle of nodes is reduced, the network can handle responding to user request and realizes a good interest satisfaction rate. We also demonstrated that our proposal realizes a gain in terms of energy. Moreover, even when the approach decides to reduce the node duty-cycle, the delay is not impacted and the packet loss ratio is not high. The combination of the two contributions enables to evaluate the energy of the proposed approach and to find a trade-off between the energy consumption and the interest satisfaction

rate.

The last contribution was focused on studying existing caching strategies and trying to find the impact of certain parameters on content placement. Then, we proposed a caching strategy that deals with content placement and replacement in the node caches. Our caching strategy CSDD '*collaborative Caching Strategy Degree and Distance aware*' choose the node on which to place the content depending on the node degree and its distance from the source node. Furthermore, a loop avoidance mechanism was implemented to overcome as possible the problem of interest loop caused by the wireless communication. After that, we carried out extensive simulation in which we varied the degree and the distance. The different proposals were compared to existing caching strategy. Findings showed that the proposed strategy overcome LCE in terms of cache hit, energy consumption and replacement rate.

In summary, this thesis provided a well-rounded set of contributions addressing content availability and dissemination when enabling content-centric networking in wireless sensor networks.

7.3 Perspectives

Although the several proposed solutions when applying CCN in WSNs, there is still room for enhancement. Indeed, the contributions of this thesis can be extended in several directions. Now, we present some of them in the following.

As short-term enhancements, in the context of data freshness and content update approaches, it would be interesting to evaluate the cost of these updates in terms of network lifetime and to set a new metric that measures the data availability hence the reliability ensured by the network. In addition to that, we are going to study the impact of the solution in different scenarios with a high network density, a bigger number of generated content and a different frequency of interest generation. Concerning the caching strategy, we will define a new metric to optimize the content placement to bypass LCE in terms of stretch and cache hit.

For the long-term perspectives, to improve our contributions, we are going to be interested in detail in the interest generation model adapted to our application scenario. We are going to study the behavior of students during the day at the campus and how interests are generated, to conduct experiments, and to investigate the real world interest generation models. A real traffic scenario will give more realistic results. Another network topology will be investigated to find the best one for the whole campus. Moreover, since the proposed energy model concerns only content forwarding, we will propose an empirical model of

the interest source. Additionally, another interesting direction for the proposed forwarding scheme is to extend our approach with a learning algorithm to help the nodes find the best threshold to reduce their duty-cycle. Due to some limitations, we plan to set some CCN functions on a small number of nodes 'super nodes' that can be either gateways or nodes with more capabilities. We will also integrate routing to try to limit the problem of broadcast in such networks. Besides, the caching strategy could be enhanced by a mechanism that considers the node and user requirements. A tradeoff could be found between the node on which to cache and the interest expectation defined by the user. Moreover, we are going to study the user mobility impact on the network performances. Finally, since we have the neOCampus platform, we will try to implement all the proposed solution on this platform to test its applicability in real scenarios. The traffic generated could also give as more idea and new perspective to enhance the proposed solutions.

List of publications

International Journals

1. G. Jaber, R. Kacimi, A. Grieco and T. Gayraud, 'An adaptive duty-cycle mechanism for energy efficient wireless sensor networks, based on information centric networking design,' In *Wireless Networks Springer Journal*, 2018, pp. 1572-8196.

International Conferences

1. G. Jaber, R. Kacimi and Z. Mammeri, 'Exploiting redundancy for energy-efficiency in Wireless Sensor Networks,' In *Wireless and Mobile Networking Conference (WMNC)*, 2016 9th IFIP pp. 180-186.
2. G. Jaber, R. Kacimi and T. Gayraud, 'Data Freshness Aware Content-Centric Networking in WSNs,' *2017 Wireless Days, Porto*, 2017, pp. 238-240.
3. G. Jaber, R. Kacimi and T. Gayraud, 'Reactive and proactive strategies for content update in content-centric and multi-users WSNs,' *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Valencia, 2017, pp. 1980-1985.
4. G. Jaber, R. Kacimi and T. Gayraud, 'Efficient Interest Satisfaction in Content Centric Wireless Sensor Networks,' In *IEEE Consumer Communications & Networking Conference (CCNC)*, 2018, Las Vegas.

Bibliographie

- [1] Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, and Moussa Ayyash. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17(4):2347–2376, 2015.
- [2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. volume 38, pages 393–422, 2004.
- [3] Publish and subscribe. <http://www.code2succeed.com/pub-sub/>. Online; accessed 15 July 2018.
- [4] Ndn specification. <http://named-data.net>. Online; accessed 15 July 2018.
- [5] Ccn stack. <https://spectrum.ieee.org/telecom/internet/a-better-way-to-organize-the-internet-contentcentric-networking>. Online; accessed 15 July 2018.
- [6] IBM. <https://www.ibm.com/internet-of-things/fr-fr/trends/>. Online; accessed 28 July 2018.
- [7] Wendi Rabiner Heinzelman, Joanna Kulik, and Hari Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. In *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom '99*, pages 174–185, New York, NY, USA, 1999. ACM.
- [8] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 8 - Volume 8, HICSS '00*, pages 8020–, Washington, DC, USA, 2000. IEEE Computer Society.

- [9] ICN. <https://irtf.org/icnrg> Online;. Online; accessed 13 August 2018.
- [10] Shuai Gao, Hongke Zhang, and Beichuan Zhang. Energy efficient interest forwarding in ndn-based wireless sensor networks. *Mobile Information Systems*, 2016, 2016.
- [11] Guoqiang Zhang, Yang Li, and Tao Lin. Caching in information centric networking: A survey. *Comput. Netw.*, 57(16):3128–3141, November 2013.
- [12] Zhong Ren, Mohamed A Hail, and Horst Hellbrück. Ccn-wsn-a lightweight, flexible content-centric networking protocol for wireless sensor networks. In *Intelligent Sensors, Sensor Networks and Information Processing, 2013 IEEE Eighth International Conference on*, pages 123–128. IEEE, 2013.
- [13] Chan Min Park, Rana Asif Rehman, and Byung-Seo Kim. Packet flooding mitigation in ccn-based wireless multimedia sensor networks for smart cities. *IEEE Access*, 5:11054–11062, 2017.
- [14] Chan Min Park, Rana Asif Rehman, and Byung-Seo Kim. Packet flooding mitigation in ccn-based wireless multimedia sensor networks for smart cities. *IEEE Access*, 5:11054–11062, 2017.
- [15] Jose Quevedo, Daniel Corujo, and Rui Aguiar. Consumer driven information freshness approach for content centric networking. In *Computer Communications Workshops (INFOCOM WKSHPS), 2014 IEEE Conference on*, pages 482–487. IEEE, 2014.
- [16] Long Cheng, Jianwei Niu, Yu Gu, Chengwen Luo, and Tian He. Achieving efficient reliable flooding in low-duty-cycle wireless sensor networks. *IEEE/ACM Transactions on Networking*, 24(6):3676–3689, 2016.
- [17] Dohyung Kim, Jong-hwan Kim, Cheoleun Moon, Jeonghwan Choi, and Ikjun Yeom. Efficient content delivery in mobile ad-hoc networks using ccn. *Ad Hoc Netw.*, 36(P1):81–99, January 2016.
- [18] G. Jaber, R. Kacimi, and T. Gayraud. Data freshness aware content-centric networking in wsns. In *2017 Wireless Days*, pages 238–240, March 2017.
- [19] Ghada Jaber, Rahim Kacimi, and Thierry Gayraud. Reactive and proactive strategies for content update in content-centric and multi-users wsns. In *IWCMC*, pages 1980–1985. IEEE, 2017.

- [20] neOCampus project. <http://www.irit/neocampus.fr>. Online; accessed 21 June 2018.
- [21] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7):1645 – 1660, 2013. Including Special sections: Cyber-enabled Distributed Computing for Ubiquitous Cloud and Network Services & Cloud Computing and Scientific Applications – Big Data, Scalable Analytics, and Beyond.
- [22] 10 technologies that will change the world in the next 10 years. <http://www.networkworld.com/article/2179278/lan-wan/10-technologies-that-will-change-the-world-in-the-next-10-years.html>. Online; accessed 17 June 2018.
- [23] Minerva Roberto, Biru Abyi, and Rotondi Domenico. Ieee iot towards definition internet of things. In *IEEE Internet of things*.
- [24] Yongrui Qin, Quan Z. Sheng, Nickolas J. G. Falkner, Schahram Dustdar, Hua Wang, and Athanasios V. Vasilakos. When things matter: A data-centric view of the internet of things. *CoRR*, abs/1407.2704, 2014.
- [25] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, pages 102–114, 2002.
- [26] Rahim Kacimi. *Techniques de conservation d'énergie pour les réseaux de capteurs sans fil*. Thèse de doctorat, Institut National Polytechnique de Toulouse, Toulouse, France, septembre 2009. (Soutenance le 28/09/2009).
- [27] Giuseppe Anastasi, Marco Conti, Mario Di Francesco, and Andrea Passarella. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Networks*, 7(3):537 – 568, 2009.
- [28] Rahim Kacimi, Riadh Dhaou, and André-Luc Beylot. Load balancing techniques for lifetime maximizing in wireless sensor networks. *Ad Hoc Networks*, 11(8):2172–2186, 2013.
- [29] Elena Fasolo, Michele Rossi, Jorg Widmer, and Michele Zorzi. In-network aggregation techniques for wireless sensor networks: a survey. *IEEE Wireless Communications*, 14(2), 2007.

- [30] Ibrahima Diane, Rahim Kacimi, Zoubir Mammeri, and Ibrahima Niang. Energy optimization in redundant wsns under deterministic and probabilistic sensing models. In *IEEE 80th Vehicular Technology Conference, VTC Fall 2014, Vancouver, BC, Canada, September 14-17, 2014*, pages 1–5, 2014.
- [31] Ghada Jaber, Rahim Kacimi, and Zoubir Mammeri. Exploiting redundancy for energy-efficiency in wireless sensor networks. In *Wireless and Mobile Networking Conference (WMNC), 2016 9th IFIP*, pages 180–186. IEEE, 2016.
- [32] Caimu Tang and Cauligi S Raghavendra. Compression techniques for wireless sensor networks. In *Wireless sensor networks*, pages 207–231. Springer, 2004.
- [33] Shashidhar Rao Gandham, Milind Dawande, Ravi Prakash, and Subbarayan Venkatesan. Energy efficient schemes for wireless sensor networks with multiple mobile base stations. In *Global telecommunications conference, 2003. GLOBECOM'03. IEEE*, volume 1, pages 377–381. IEEE, 2003.
- [34] Hyewon Jun, Wenrui Zhao, Mostafa H Ammar, Ellen W Zegura, and Chungki Lee. Trading latency for energy in wireless ad hoc networks using message ferrying. In *Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference on*, pages 220–225. IEEE, 2005.
- [35] Z.J. Haas, J.Y. Halpern, and Li Li. Gossip-based ad hoc routing. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1707–1716 vol.3, 2002.
- [36] Mallanagouda Patil and Rajashekhar C. Biradar. A survey on routing protocols in wireless sensor networks. *2012 18th IEEE International Conference on Networks (ICON)*, pages 86–91, 2012.
- [37] R. Rajagopalan and P. K. Varshney. Data-aggregation techniques in sensor networks: A survey. *Commun. Surveys Tuts.*, 8(4):48–63, October 2006.
- [38] MJ Handy, Marc Haase, and Dirk Timmermann. Low energy adaptive clustering hierarchy with

- deterministic cluster-head selection. In *Mobile and Wireless Communications Network, 2002. 4th International Workshop on*, pages 368–372. IEEE, 2002.
- [39] J. Grover, Shikha, and M. Sharma. Location based protocols in wireless sensor network – a review. In *2014 5th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, volume 00, pages 1–5, July 2014.
- [40] Yan Yu, Ramesh Govindan, and Deborah Estrin. Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks. 2001.
- [41] Sinchan Roychowdhury and Chiranjib Patra. Geographic adaptive fidelity and geographic energy aware routing in ad hoc routing. In *International Conference*, volume 1, pages 309–313, 2010.
- [42] R. Sumathi and M. G. Srinivas. A survey of qos based routing protocols for wireless sensor networks. *JIPS*, 8:589–602, 2012.
- [43] S. Archana and A. S. Salvan. Sar protocol based secure data aggregation in wireless sensor network. In *2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO)*, pages 1–6, Jan 2015.
- [44] K. Akkaya and M. Younis. An energy-aware qos routing protocol for wireless sensor networks. In *23rd International Conference on Distributed Computing Systems Workshops, 2003. Proceedings.*, pages 710–715, May 2003.
- [45] Luwei Jing, Feng Liu, and Yuling Li. Energy saving routing algorithm based on spin protocol in wsn. In *2011 International Conference on Image Analysis and Signal Processing*, pages 416–419, Oct 2011.
- [46] M. M. Warriar and A. Kumar. Energy efficient routing in wireless sensor networks: A survey. In *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pages 1987–1992, March 2016.
- [47] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann, and Fabio Silva. Directed diffusion for wireless sensor networking. *IEEE/ACM Trans. Netw.*, 11(1):2–16, February 2003.

- [48] BPS Sahoo and Deepak Puthal. Drug: An energy-efficient data-centric routing protocol for wireless sensor networks. *arXiv preprint arXiv:1404.4685*, 2014.
- [49] Luca Mainetti, Luigi Patrono, and Antonio Vilei. Evolution of wireless sensor networks towards the internet of things: A survey. In *Software, Telecommunications and Computer Networks (SoftCOM), 2011 19th International Conference on*, pages 1–6. IEEE, 2011.
- [50] Delphine Christin, Andreas Reinhardt, Parag S. Mogre, and Ralf Steinmetz. Wireless sensor networks and the internet of things: Selected challenges.
- [51] Gyan Prakash Mishra and Mayank Dave. A review on content centric networking and caching strategies. In *Communication Systems and Network Technologies (CSNT), 2015 Fifth International Conference on*, pages 925–929. IEEE, 2015.
- [52] Bhaskar Krishnamachari, Deborah Estrin, and Stephen Wicker. Modelling data-centric routing in wireless sensor networks. In *IEEE infocom*, volume 2, pages 39–44, 2002.
- [53] George Xylomenos, Christopher N Ververidis, Vasilios A Siris, Nikos Fotiou, Christos Tsilopoulos, Xenofon Vasilakos, Konstantinos V Katsaros, and George C Polyzos. A survey of information-centric networking research. *IEEE Communications Surveys & Tutorials*, 16(2):1024–1049.
- [54] Dipankar Raychaudhuri, Kiran Nagaraja, and Arun Venkataramani. Mobilityfirst: A robust and trustworthy mobility-centric architecture for the future internet. *SIGMOBILE Mob. Comput. Commun. Rev.*, 16(3):2–13, December 2012.
- [55] Ashok Anand, Fahad Dogar, Dongsu Han, Boyan Li, Hyeontaek Lim, Michel Machado, Wenfei Wu, Aditya Akella, David G. Andersen, John W. Byers, Srinivasan Seshan, and Peter Steenkiste. Xia: An architecture for an evolvable and trustworthy internet. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks, HotNets-X*, pages 2:1–2:6, New York, NY, USA, 2011. ACM.
- [56] Van Jacobson, Diana K Smetters, James D Thornton, Michael F Plass, Nicholas H Briggs, and Rebecca L Braynard. Networking named content. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 1–12. ACM, 2009.

- [57] CCNx. <http://blogs.parc.com/ccnx/>; Online; accessed 18 July 2018.
- [58] Maroua Meddeb, Amine Dhraief, Abdelfettah Belghith, Thierry Monteil, Khalil Drira, and Saad AlAhmadi. Cache freshness in named data networking for the internet of things. *The Computer Journal*, 2018.
- [59] Bohao Feng, Huachun Zhou, Hongke Zhang, Jiaojiao Jiang, and Shui Yu. A popularity-based cache consistency mechanism for information-centric networking. In *Global Communications Conference (GLOBECOM), 2016 IEEE*, pages 1–6. IEEE, 2016.
- [60] Bengt Ahlgren, Christian Dannewitz, Claudio Imbrenda, Dirk Kutscher, and Borje Ohlman. A survey of information-centric networking. *IEEE Communications Magazine*, 50(7), 2012.
- [61] Athanasios V. Vasilakos, Zhe Li, Gwendal Simon, and Wei You. Information centric network: Research challenges and opportunities. *Journal of Network and Computer Applications*, 52:1 – 10, 2015.
- [62] M. S. Akbar, K. A. Khaliq, R. N. B. Rais, and A. Qayyum. Information-centric networks: Categorizations, challenges, and classifications. In *2014 23rd Wireless and Optical Communication Conference (WOCC)*, pages 1–5, May 2014.
- [63] Claude Chaudet, Isabelle Demeure, Salma Ktari, Nicola Costagliola, and Samuel Tardieu. Publish/subscribe for wireless sensor networks. In *Proceedings of the 7th Asian Internet Engineering Conference*, pages 19–21. ACM, 2011.
- [64] Divya Saxena, Vaskar Raychoudhury, Neeraj Suri, Christian Becker, and Jiannong Cao. Named data networking: A survey. *Computer Science Review*, 19:15 – 55, 2016.
- [65] Lan Wang, AKMM Hoque, Cheng Yi, Adam Alyyan, and Beichuan Zhang. Ospfn: An ospf based routing protocol for named data networking. Technical report, Technical Report NDN-0003, 2012.
- [66] Huichen Dai, Jianyuan Lu, Yi Wang, and Bin Liu. A two-layer intra-domain routing scheme for named data networking. In *2012 IEEE Global Communications Conference (GLOBECOM)*, pages 2815–2820, Dec 2012.

- [67] AKM Hoque, Syed Obaid Amin, Adam Alyyan, Beichuan Zhang, Lixia Zhang, and Lan Wang. Nlsr: named-data link state routing protocol. In *Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking*, pages 15–20. ACM, 2013.
- [68] Jihoon Lee and Daeyoub Kim. Proxy-based mobility management scheme in mobile content centric networking (ccn) environments. *IEEE Transactions on Consumer Electronics*, 57(2), 2011.
- [69] Van Jacobson, Diana K. Smetters, James D. Thornton, Michael F. Plass, Nicholas H. Briggs, and Rebecca L. Braynard. Networking named content. In *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, CoNEXT '09*, pages 1–12, New York, NY, USA, 2009. ACM.
- [70] Bilel Saadallah, Abdelkader Lahmadi, and Olivier Festor. *CCNx for Contiki: implementation details*. PhD thesis, INRIA, 2012.
- [71] C. Severance. Van jacobson: Content-centric networking. *Computer*, 46(1):11–13, Jan 2013.
- [72] Ngoc-Thanh Dinh and Younghun Kim. Potential of information-centric wireless sensor and actor networking. In *Computing, Management and Telecommunications (ComManTel), 2013 International Conference on*, pages 163–168. IEEE, 2013.
- [73] Marica Amadeo, Claudia Campo, Antonella Molinaro, and Nathalie Mitton. Named Data Networking: a Natural Design for Data Collection in Wireless Sensor Networks. In *IEEE Wireless Days (WD)*, Valencia, Spain, November 2013.
- [74] Jan Pieter Meijers, Marica Amadeo, Claudia Campolo, Antonella Molinaro, Stefano Yuri Paratore, Giuseppe Ruggeri, and Marthinus J Booyesen. A two-tier content-centric architecture for wireless sensor networks. In *Network Protocols (ICNP), 2013 21st IEEE International Conference on*, pages 1–2. IEEE, 2013.
- [75] Naveen Chauhan, LK Awasthi, and Narottam Chand. Global cooperative caching for wireless sensor networks. In *Information and Communication Technologies (WICT), 2011 World Congress on*, pages 235–239. IEEE, 2011.

- [76] Gayathri Tilak Singh and Fadi M. Al-Turjman. A data delivery framework for cognitive information-centric sensor networks in smart outdoor monitoring. *Computer Communications*, 74:38 – 51, 2016. Current and Future Architectures, Protocols, and Services for the Internet of Things.
- [77] Ahmed Aboud and Haïfa Touati. Geographic interest forwarding in ndn-based wireless sensor networks. In *Computer Systems and Applications (AICCSA), 2016 IEEE/ACS 13th International Conference of*, pages 1–8. IEEE, 2016.
- [78] Bo Chen, Liang Liu, Zhao Zhang, Wenbo Yang, and Huadong Ma. Brr-cvr: A collaborative caching strategy for information-centric wireless sensor networks. In *Mobile Ad-Hoc and Sensor Networks (MSN), 2016 12th International Conference on*, pages 31–38. IEEE, 2016.
- [79] Cesar Ghali, Marc A. Schlosberg, Gene Tsudik, and Christopher A. Wood. Interest-based access control for content centric networks. In *Proceedings of the 2Nd ACM Conference on Information-Centric Networking*, ACM-ICN '15, pages 147–156, New York, NY, USA, 2015. ACM.
- [80] Otto Waltari and Jussi Kangasharju. Content-centric networking in the internet of things. In *Consumer Communications & Networking Conference (CCNC), 2016 13th IEEE Annual*, pages 73–78. IEEE, 2016.
- [81] Marie-Aurelie Nef, Leonidqs Perlepes, Sophia Karagiogou, and Panayotis K. Kikiras. Enabling qos in the internet of things. *CTRQ*, 2012.
- [82] J. Rego, D. Corujo, and R. Aguiar. Consumer driven information freshness approach for content centric networking. In *IEEE The 2nd Workshop on Name Oriented Mobility - INFOCOM 2014 - NOM*, volume -, pages 482–487, April 2014.
- [83] SM Iddalagi and SK Mahabaleshwar. Dynamic data forwarding in wireless sensor networks. In *Broadband and Wireless Computing, Communication and Applications (BWCCA), 2013 Eighth International Conference on*, pages 506–511. IEEE, 2013.
- [84] Yu Gu and Tian He. Dynamic switching-based data forwarding for low-duty-cycle wireless sensor networks. *IEEE Transactions on Mobile Computing*, 10(12):1741–1754, 2011.

- [85] Marica Amadeo, Claudia Campolo, and Antonella Molinaro. Multi-source data retrieval in iot via named data networking. In *Proceedings of the 1st ACM Conference on Information-Centric Networking*, ACM-ICN '14, pages 67–76, New York, NY, USA, 2014. ACM.
- [86] Emmanuel Baccelli, Christian Mehlis, Oliver Hahm, Thomas C. Schmidt, and Matthias Wählisch. Information centric networking in the iot: Experiments with ndn in the wild. In *Proceedings of the 1st ACM Conference on Information-Centric Networking*, ACM-ICN '14, pages 77–86, New York, NY, USA, 2014. ACM.
- [87] Yanhua Li, Haiyong Xie, Yonggang Wen, and Zhi-Li Zhang. Coordinating in-network caching in content-centric networks: Model and analysis. In *Proceedings of the 2013 IEEE 33rd International Conference on Distributed Computing Systems*, ICDCS '13, pages 62–72, Washington, DC, USA, 2013. IEEE Computer Society.
- [88] John Dilley, Bruce Maggs, Jay Parikh, Harald Prokop, Ramesh Sitaraman, and Bill Weihl. Globally distributed content delivery. *IEEE Internet Computing*, 6(5):50–58, September 2002.
- [89] Stefan Podlipnig and Laszlo Böszörményi. A survey of web cache replacement strategies. *ACM Comput. Surv.*, 35(4):374–398, December 2003.
- [90] Xiong Wang, Jing Ren, Tong Tong, Rui Dai, Shizhong Xu, and Sheng Wang. Towards efficient and lightweight collaborative in-network caching for content centric networks. In *Global Communications Conference (GLOBECOM), 2016 IEEE*, pages 1–7. IEEE, 2016.
- [91] Konstantinos Katsaros, George Xylomenos, and George C Polyzos. Multicache: An overlay architecture for information-centric networking. *Computer Networks*, 55(4):936–947, 2011.
- [92] Mokrane Bouzeghoub. A framework for analysis of data freshness. In *Proceedings of the 2004 International Workshop on Information Quality in Information Systems*, IQIS '04, pages 59–67, New York, NY, USA, 2004. ACM.
- [93] Ignacio Solis and Katia Obraczka. In-network aggregation trade-offs for data collection in wireless sensor networks. *Int. J. Sen. Netw.*, 1(3/4):200–212, January 2006.

- [94] Shehzad Ashraf Ch, Mian Muhammad Omair, Iftikhar Ali Khan, and Tahir Afzal Malik. Ensuring reliability and freshness for data aggregation in wireless sensor networks. *International Journal of Machine Learning and Computing*, 1(3):224, 2011.
- [95] Ramesh Rajagopalan and Pramod K Varshney. Data aggregation techniques in sensor networks: A survey. 2006.
- [96] Triana Mugia Rahayu, Sang-Gon Lee, and Hoon-Jae Lee. A secure routing protocol for wireless sensor networks considering secure data aggregation. *Sensors*, 15(7):15127–15158, 2015.
- [97] Chen Min, Leung Victor C. M., Mao Shiwen, and Kwon Taekyoung. Receiver-oriented load-balancing and reliable routing in wireless sensor networks. *Wireless Communications and Mobile Computing*, 9(3):405–416.
- [98] Jamal N Al-Karaki and Ahmed E Kamal. Routing techniques in wireless sensor networks: a survey. *IEEE wireless communications*, 11(6):6–28, 2004.
- [99] A. Sharma and P. Lakkadwala. Performance comparison of reactive and proactive routing protocols in wireless sensor network. In *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization*, pages 1–6, Oct 2014.
- [100] A. Kolioussis and J. Sventek. Proactive vs reactive routing for wireless sensor networks. Technical report, University of Glasgow, Glasgow, UK, 2007.
- [101] Rahul C Shah and Jan M Rabaey. Energy aware routing for low energy ad hoc sensor networks. In *Wireless Communications and Networking Conference, 2002. WCNC2002. 2002 IEEE*, volume 1, pages 350–355. IEEE, 2002.
- [102] A. Manjeshwar and D. P. Agrawal. Teen: a routing protocol for enhanced efficiency in wireless sensor networks. In *Proceedings 15th International Parallel and Distributed Processing Symposium. IPDPS 2001*, pages 2009–2015, April 2001.
- [103] M. Gallo, D. Perino, and L. Muscariello. Content-centric networking packet header format draft-ccn-packet-header-00. *RFC*, pages 1–16, 2015.

- [104] D. Kutscher, K. Pentikousis, D. Saucez, and T. Schmidt. Information-centric networking (icn) research challenges. *RFC*, 7927:1–37, 2016.
- [105] H. Asaeda and X. Shaos and. Ccninfo: Discovering content and network information in content-centric networks. *RFC*, pages 1–33, 2018.
- [106] Marcel Steine, Cuong Viet Ngo, Ramon Serna Oliver, Marc Geilen, Twan Basten, Gerhard Fohler, and Jean-Dominique Decotignie. Proactive reconfiguration of wireless sensor networks. In *Proceedings of the 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '11, pages 31–40, New York, NY, USA, 2011. ACM.
- [107] Contiki. <http://www.contiki-os.org>. Online, accessed 16 August 2018.
- [108] S. M. Lasassmeh and J. M. Conrad. Time synchronization in wireless sensor networks: A survey. In *Proceedings of the IEEE SoutheastCon 2010 (SoutheastCon)*, pages 242–245, March 2010.
- [109] Ghada Jaber, Rahim Kacimi, Luigi Alfredo Grieco, and Thierry Gayraud. An adaptive duty-cycle mechanism for energy efficient wireless sensor networks, based on information centric networking design. *Wireless Networks*, Sep 2018.
- [110] Ricardo C Carrano, Diego Passos, Luiz CS Magalhaes, and Celio VN Albuquerque. Survey and taxonomy of duty cycling mechanisms in wireless sensor networks. *IEEE Communications Surveys & Tutorials*, 16(1):181–194, 2014.
- [111] C. M. Vigorito, D. Ganesan, and A. G. Barto. Adaptive control of duty cycling in energy-harvesting wireless sensor networks. In *2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 21–30, June 2007.
- [112] Feng Wang and Jiangchuan Liu. On reliable broadcast in low duty-cycle wireless sensor networks. *IEEE Transactions on Mobile Computing*, 11(5):767–779, 2012.
- [113] M Riduan Ahmad, Eryk Dutkiewicz, Xiaojing Huang, et al. A survey of low duty cycle mac protocols in wireless sensor networks. 2011.

- [114] Jyoti Saraswat and Partha Pratim Bhattacharya. Effect of duty cycle on energy consumption in wireless sensor networks. *International Journal of Computer Networks & Communications*, 5(1):125, 2013.
- [115] Nassima Bouadem, Rahim Kacimi, and Abdelkamel Tari. Impact of duty-cycling: Towards mostly-off sensor networks. *Int. J. Bus. Data Commun. Netw.*, 12(1):16–35, January 2016.
- [116] Anthony Rowe, Rahul Mangharam, and Raj Rajkumar. Rt-link: A global time-synchronized link protocol for sensor networks. *Ad Hoc Networks*, 6(8):1201 – 1220, 2008. Energy Efficient Design in Wireless Ad Hoc and Sensor Networks.
- [117] Gang Lu, Bhaskar Krishnamachari, and Cauligi S Raghavendra. An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks. In *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, page 224. IEEE, 2004.
- [118] Wei Ye, John Heidemann, and Deborah Estrin. An energy-efficient mac protocol for wireless sensor networks. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1567–1576. IEEE, 2002.
- [119] Tijs van Dam and Koen Langendoen. An adaptive energy-efficient mac protocol for wireless sensor networks. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, SenSys '03*, pages 171–180, New York, NY, USA, 2003. ACM.
- [120] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems, SenSys '04*, pages 95–107, New York, NY, USA, 2004. ACM.
- [121] Amre El-Hoiydi and Jean-Dominique Decotignie. Wisemac: An ultra low power mac protocol for multi-hop wireless sensor networks. In Sotiris E. Nikolettseas and José D. P. Rolim, editors, *Algorithmic Aspects of Wireless Sensor Networks*, pages 18–31, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

- [122] Yanjun Sun, Omer Gurewitz, and David B. Johnson. Ri-mac: A receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, SenSys '08*, pages 1–14, New York, NY, USA, 2008. ACM.
- [123] Vamsi Paruchuri, Shivakumar Basavaraju, Arjan Durresi, Rajgopal Kannan, and S Sitharama Iyengar. Random asynchronous wakeup protocol for sensor networks. In *Broadband Networks, 2004. BroadNets 2004. Proceedings. First International Conference on*, pages 710–717. IEEE, 2004.
- [124] Kyle Guan, Gary Atkinson, Daniel C Kilper, and Ece Gulsen. On the energy efficiency of content delivery architectures. In *Communications Workshops (ICC), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.
- [125] Nakjung Choi, Kyle Guan, Daniel C Kilper, and Gary Atkinson. In-network caching effect on optimal energy consumption in content-centric networking. In *Communications (ICC), 2012 IEEE International Conference on*, pages 2889–2894. IEEE, 2012.
- [126] J. Li, B. Liu, and H. Wu. Energy-efficient in-network caching for content-centric networking. *IEEE Communications Letters*, 17(4):797–800, April 2013.
- [127] Mianxiong Dong, Kaoru Ota, Anfeng Liu, and Minyi Guo. Joint optimization of lifetime and transport delay under reliability constraint wireless sensor networks. *IEEE Transactions on Parallel & Distributed Systems*, (1):1–1, 2016.
- [128] Pratyay Kuila and Prasanta K. Jana. Energy efficient clustering and routing algorithms for wireless sensor networks: Particle swarm optimization approach. *Engineering Applications of Artificial Intelligence*, 33:127 – 140, 2014.
- [129] Seyed Kaveh Fayazbakhsh, Yin Lin, Amin Tootoonchian, Ali Ghodsi, Teemu Koponen, Bruce Maggs, K.C. Ng, Vyas Sekar, and Scott Shenker. Less pain, most of the gain: Incrementally deployable icn. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM, SIGCOMM '13*, pages 147–158, New York, NY, USA, 2013. ACM.

- [130] A. K. Mahani and H. F. Rashvand. Novel sensor distribution for improved networking performance. In *IET International Conference on Wireless Sensor Network 2010 (IET-WSN 2010)*, pages 331–336, Nov 2010.
- [131] Sarika Yadav and Rama Shankar Yadav. A review on energy efficient protocols in wireless sensor networks. *Wireless Networks*, 22(1):335–350, Jan 2016.
- [132] Jayant Baliga, Robert Ayre, Kerry Hinton, and Rodney S Tucker. Architectures for energy-efficient iptv networks. In *Optical Fiber Communication-includes post deadline papers, 2009. OFC 2009. Conference on*, pages 1–3. IEEE, 2009.
- [133] Hao Che, Ye Tung, and Zhijun Wang. Hierarchical web caching systems: modeling, design and experimental results. *IEEE Journal on Selected Areas in Communications*, 20(7):1305–1314, 2002.
- [134] Tmote Sky Datasheet. <http://www.eecs.harvard.edu/tmote-sky-datasheet.pdf>. Online; accessed 22 June 2018.
- [135] Yu Gu and Tian He. Dynamic switching-based data forwarding for low-duty-cycle wireless sensor networks. *IEEE Transactions on Mobile Computing*, 10(12):1741–1754, 2011.
- [136] Hashim A. Hashim, B.O. Ayinde, and M.A. Abido. Optimal placement of relay nodes in wireless sensor network using artificial bee colony algorithm. *J. Netw. Comput. Appl.*, 64(C):239–248, April 2016.
- [137] Sudhanshu Pant, Naveen Chauhan, and Prashant Kumar. Effective cache based policies in wireless sensor networks: A survey. 2010.
- [138] Nikos Dimokas, Dimitrios Katsaros, Leandros Tassioulas, and Yannis Manolopoulos. High performance, low complexity cooperative caching for wireless sensor networks. volume 17, pages 717–737. Springer, 2011.
- [139] N. Dimokas, D. Katsaros, L. Tassioulas, and Y. anolopoulos. High performance, low complexity cooperative caching for wireless sensor networks. In *2009 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks Workshops*, pages 1–9, June 2009.

- [140] Narottam Chand. Cooperative data caching in wsn. *World Academy of Science, Engineering and Technology*, 63:90–94, 2012.
- [141] Lorenzo Saino. *On the Design of Efficient Caching Systems*. PhD thesis, University College London, 2015.
- [142] A. Suki, M. ArifSuhaidi, and HassanSuhaidi HassanIbrahim AbdullahiIbrahim Abdullahi. Cache replacement positions in information-centric network. In *International Conference on Internet Applications, Protocol and Services (NETAPPS2015)*, Oct. 2015.
- [143] Wei Koong Chai, Diliang He, Ioannis Psaras, and George Pavlou. Cache less for more in information-centric networks (extended version). *Computer Communications*, 36(7):758–770, 2013.
- [144] K. Suksomboon, S. Tarnoi, Yusheng Ji, M. Koibuchi, K. Fukuda, S. Abe, N. Motonori, M. Aoki, S. Urushidani, and S. Yamada. Popcache: Cache more or less based on content popularity for information-centric networking. In *2013 IEEE 38th Conference on Local Computer Networks (LCN 2013)(LCN)*, volume 00, pages 236–243, Oct. 2014.
- [145] Sagnik Bhattacharya, Hyung Kim, Shashi Prabh, and Tarek Abdelzaher. Energy-conserving data placement and asynchronous multicast in wireless sensor networks. In *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, MobiSys '03*, pages 173–185, New York, NY, USA, 2003. ACM.
- [146] K. Shashi Prabh and Tarek F. Abdelzaher. Energy-conserving data cache placement in sensor networks. *ACM Trans. Sen. Netw.*, 1(2):178–203, November 2005.
- [147] Junaid A. Khan, Cedric Westphal, and Yacine Ghamri-Doudane. A Popularity-aware Centrality Metric for Content Placement in Information Centric Networks. In *ICNC 2018 - International Conference on Computing, Networking and Communication*, Maui, Hawaii, United States, March 2018.
- [148] Report IRIT. <https://www.irit.fr/neocampus/fr/>. Online; accessed 15 September 2018.
- [149] C. Rosenberg, V. P. Mhatre, N. Shroff, D. Kofman, and R. Mazumdar. A minimum cost heterogeneous sensor network with a lifetime constraint. *IEEE Transactions on Mobile Computing*, 4:4–15, 01 2005.

- [150] César Bernardini, Thomas Silverston, and Olivier Festor. A comparison of caching strategies for content centric networking. In *Global Communications Conference (GLOBECOM), 2015 IEEE*, pages 1–6. IEEE, 2015.