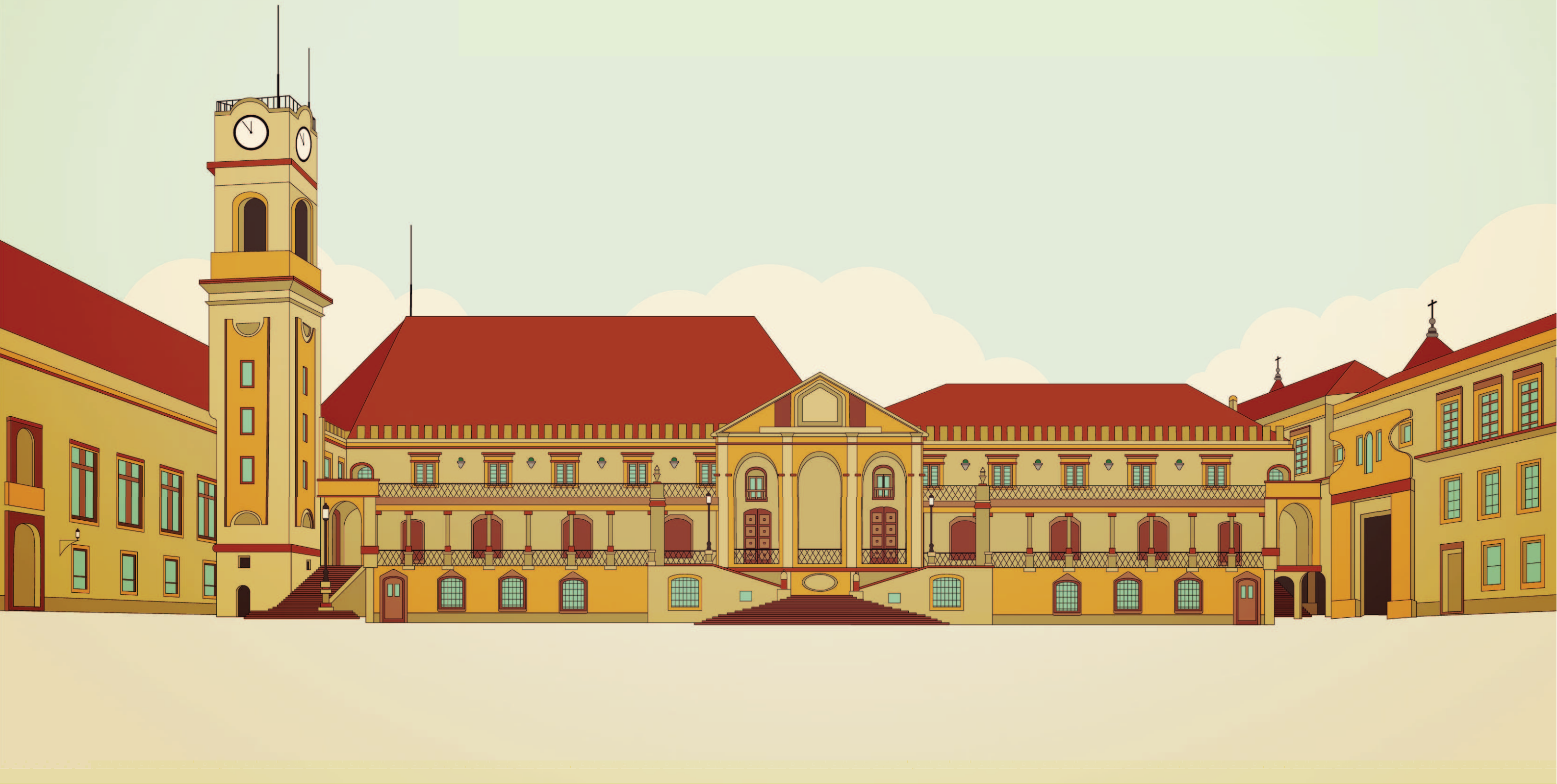


André Figueira Riker

Group communication mechanisms in Machine-to-Machine environments

Tese de Doutoramento do Programa de Doutoramento em Ciências e Tecnologias da Informação, orientada pela Professora Doutora Marilia Curado e pelo Professor Doutor Edmundo Monteiro e apresentada ao Departamento de Engenharia Informática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Julho de 2018

UNIVERSIDADE DE COIMBRA

Department of Informatics Engineering
Faculty of Sciences and Technology
University of Coimbra

Group communication mechanisms in Machine-to-Machine environments

André Figueira Riker

Doctoral Program in Information Science and Technology
PhD Thesis submitted to the University of Coimbra

Advised by Dr. Marilia Pascoal Curado
and by Prof. Dr. Edmundo Monteiro

July, 2018

· U C ·

Departamento de Engenharia Informática
Faculdade de Ciências e Tecnologia
Universidade de Coimbra

Mecanismos de Comunicação em Grupo em Ambientes Máquina para Máquina

André Figueira Riker

Programa de Doutoramento em Ciências e Tecnologias da Informação
Tese de Doutoramento apresentada à Universidade de Coimbra

Orientado pela Prof. Dra. Marilia Pascoal Curado
e pelo Prof. Dr. Edmundo Monteiro

Julho de 2018

This work was partially supported by Portuguese Foundation for Science and Technology (FCT) under the grant MORFEU (PTDC/EEA-CRO/108348/2008) and DENSENET (PTDC/EEI-SCR/6453/2014), as well as by Brazilian National Counsel of Technological and Scientific Development (CNPq) under the grant 246599/2012-0.

To my dear grandparents, Lene & Sinval Figueira and Rosa & Delmas Riker.
Even though most of you are not here, your legacy is everlasting.

Aos meus queridos avós, Lene & Sinval Figueira e Rosa & Delmas Riker.
Mesmo que a maioria de vocês não esteja aqui, o vosso legado é eterno.

Acknowledgements

During all these long six years, me and my wife Andressa have experienced many arrivals and farewells. It seems that God has lead us to mountaintops and valleys. After this long journey, we can look back and see a lot of inspiration and fruits on our beings. Thanks God.

I need to say more than thanks to someone who embraced this journey since it was a distant dream. Someone who quit her job, left friends and family at her soo loved place to follow a crazy idea. At the end of this life fragment, I see how I am a blessed guy to be married to you, Andressa. I will never forget all you have done for us.

I say us because now we have our little tuga baby, Clarisse. So, I have to say thanks to Clarisse, a gift from God, and also to all the coming little, beautiful, noisy, loved babies. You are my motivation.

A special thanks goes to my family, which I left in Brazil and I miss soo much. My mother Márcia, my father Delmas Riker, everything I know originated from your invaluable teachings. No one loves me like you do, unconditionally. Thank you, my brothers Daniel, David and Francisco. I cannot forget, Brena and Taynah.

My thanks also go to my PhD supervisors, Marília and Edmundo. I appreciate all the effort you dedicated to me during these years. I am really grateful for the opportunity you gave me to grow with your advice.

I need to mention a special person, Eduardo Cerqueira. If I am finishing this PhD is because this professor and friend believed I could. Meeting you, back in 2010, was the turning point for this journey.

Some people may not know, but we have found a family in Portugal. People who truly care about us. Pastor Marco Carvalho, Néia, Joãozinho, Sarinha, Lili, maestro Eliseu Barros, Pedro, Katiuce, Dona Isabel, seu Antônio, dona Florinda and many others friends. Without you, our life in Coimbra would not be the same.

Last but not least, Bruno Dalmazo and Roger Immich. Without you I would not had laughed and drunk so much coffee as I did.

Abstract

Machine-to-Machine (M2M) is an emerging paradigm that rapidly has gained ground in the scenario of modern wireless communications. In recent years, companies have started to introduce numerous products and services for machine-to-machine communication, and it is still expected that M2M will enable the connectivity of millions of devices. The central idea of this paradigm is to provide a pervasive presence of machines and make them share data cooperatively to take decisions with low human intervention. In this context, group communication is present in M2M as a type of communication frequently required, since the number of devices is expected to be large. However, the complexity, the heterogeneity, the large scale, and vast types of scenarios turn efficient machine-to-machine group communication into a major challenge.

Tackling this challenge, this thesis in the first part proposes mechanisms, usually classified as in-network data aggregation, to improve machine-to-machine group communication by allowing the member nodes of a M2M group to eliminate data redundancy and to reduce overhead caused by transmissions of small payloads. Additionally to the proposed in-network data aggregation, this thesis presents tree-based heuristics and also formulates a mathematical model to compute the maximum network lifetime. It also discusses a performance comparison between the proposed heuristics and the optimal solution, highlighting the pros and cons of each approach. Furthermore, this thesis describes a set of experiments using real devices to test the feasibility of the proposals and measure the performance of the improvements in a group communication scenario.

The second part of this thesis addresses the problem of energy efficiency of nodes with capabilities to harvest energy from the ambient. Regarding this problem, two mechanisms that apply data aggregation procedures are proposed. In both cases, the applied data aggregation procedures are used to regulate the amount of traffic flowing over the paths. The ultimate objective of this traffic regulation is to control the energy consumption of the nodes in order to achieve neutral operation or energy sustainability. The condition to be in neutral operation is to keep the remaining energy of a node away from maximum and minimum. The first mechanism uses the residual energy as parameter to determine the amount of data that should be aggregated before the transmission. The second mechanism exploits the fact that there is a bottleneck in terms of residual energy in the path towards the gateway. This bottleneck is the node with minimum energy reserve in the path. So, the enhanced mechanism exchanges messages to find the critical node which is causing the bottleneck and aggregates the traffic to maintain the critical node in neutral operation. This mechanism is implemented in a real embedded system and adapts and extends some of the M2M standard protocols.

The benefits and drawbacks of the contributions have been demonstrated using different evaluation methods, such as simulations, mathematical models, and real experimentation. Furthermore, the evaluations have been conducted with well-known metrics and the results show that all the proposed mechanisms were able to outperform related proposals in literature.

Keywords: Machine-to-Machine (M2M); Data Aggregation; Energy Efficiency; Network Lifetime; Neutral Operation.

Resumo

Máquina para máquina(M2M) é um paradigma emergente que rapidamente ganhou terreno no cenário das comunicações sem fio modernas. Nos últimos anos, as empresas começaram a introduzir inúmeros produtos e serviços para a comunicação M2M, e ainda é esperado que M2M permita a conectividade de milhares e milhões de dispositivos. A idéia central desse paradigma é fornecer presença pervasiva de dispositivos e fazê-los compartilhar dados de forma cooperativa para tomar decisões com baixa intervenção humana. Neste contexto, a comunicação em grupo está presente em M2M como um tipo de comunicação frequentemente necessária, uma vez que se espera que o número de dispositivos seja grande. No entanto, a complexidade, a heterogeneidade, a grande escala e os vastos tipos de cenários transformam a comunicação eficiente de grupos M2M em um grande desafio.

Enfrentando este desafio, na primeira parte desta tese é proposto procedimentos para melhorar a comunicação em grupo M2M, permitindo que o nó membro de um grupo M2M elimine a redundância de dados e reduza o overhead causado por transmissões de pequenos payloads. Adicionalmente aos procedimentos propostos, esta tese apresenta heurísticas baseadas em árvore e também formula um modelo matemático para calcular o tempo de vida máximo da rede. Uma comparação de desempenho entre as heurísticas propostas e a solução ótima também é discutida, destacando-se os prós e contras de cada abordagem. Além disso, esta tese descreve um conjunto de experimentos usando dispositivos reais que testam a viabilidade e medem o desempenho das melhorias propostas em um cenário de comunicação em grupo.

A segunda parte da tese aborda o problema da eficiência energética de nós com capacidade de extrair energia do ambiente. Em relação a este problema, dois mecanismos que aplicam agregação de dados são propostos. Em ambos casos, os procedimentos de agregação de dados são usados para regular a quantidade de tráfego que fluem pelos caminhos. O objetivo final de regular o tráfego é controlar o consumo de energia dos nós a fim de alcançar operação neutra ou sustentabilidade energética. A condição para estar em operação neutra significa manter a energia residual da reserva dentro um intervalo longe do máximo e mínimo. O primeiro mecanismo proposto com esse fim usa a energia residual como parâmetro para determinar a quantidade de dados que deve ser agregada antes da transmissão. O segundo mecanismo explora o facto que existe um gargalo em termos de energia residual no caminho em direção ao gateway. Esse gargalo é o nó com reserva mínima de energia no caminho. Assim, o mecanismo aprimorado troca mensagens para encontrar nó crítico que o gargalo de energia no caminho e agrega o tráfego para manter o nó crítico em operação neutra. Este mecanismo foi implementado em um sistema embarcado real, adapta e estende alguns protocolos padrões de M2M.

Os benefícios e desvantagens das contribuições foram demonstrados usando diferentes métodos de avaliação, tais como simulações, modelos matemáticos e experimentação real. Além disso, as avaliações foram realizadas com métricas bem conhecidas e os resultados mostram que todos os mecanismos propostos foram capazes de superar propostas relacionadas da literatura.

Palavras-chave: Máquina para Máquina (M2M); Agregação de Dados; Eficiência Energética; Vida Útil da Rede; Operação Neutra.

Foreword

The work detailed in this thesis was accomplished at the Laboratory of Communication and Telematics (LCT) of the Centre for Informatics and Systems of the University of Coimbra (CISUC), within the context of the following projects:

Project MORFEU - Portuguese Foundation for Science and Technology (FCT) MORFEU project (PTDC/EEA-CRO/108348/2008).

Project DENSENET - Efficient Communication in Dense Networks (DENSENET) project (PTDC/EEI- SCR/6453/2014), co-financed by FCT/FEDER/COMPETE 2020 project.

This work was funded by the following grants:

Project grant Portuguese Foundation for Science and Technology (FCT) MORFEU project (PTDC/EEA-CRO/108348/2008).

Doctoral grant Brazilian National Counsel of Technological and Scientific Development (CNPq) (246599/2012-0).

Project grant Portuguese Foundation for Science and Technology (FCT) DENSENET project (PTDC/EEI-SCR/6453/2014).

The outcome of the design, experiments, and assessments of several mechanisms on the course of this work resulted in the following publications:

Journal papers:

- Riker, A. and Cerqueira, E. and Curado, M. and Monteiro, E., "**A Two-Tier Adaptive Data Aggregation Approach for M2M Group-Communication**", Sensors Journal, IEEE, Volume: 16, Issue: 3, pp 823-835, IEEE, 2016. **Impact Factor: 2.512**.
- Riker, A. and Fonseca, C. and Curado, M. and Monteiro, E., "**Energy-efficient multigroup communication**", Transactions on Emerging Telecommunications Technologies, Wiley, Volume: 29, Issue: 3, pp 1-15, IEEE, 2017. **Impact Factor: 1.535**.

Book chapters:

- Riker, A. and Curado, M. and Monteiro, E., "**Role Coordination in Large-Scale and Highly-Dense Internet-of-Things**", in Internet of

Things (IoT) in 5G Mobile Technologies, Volume 8, pp 77-98, Springer, 2016.

- Riker, A. and Curado, M. and Monteiro, E., "**Group communication in machine-to-machine environments**", in Wireless Networking for Moving Objects, Volume 8611, pp 219-238, Springer LNCS, 2014.

Conference papers:

- Riker, A. and Granjal, J. and Curado, M. and Monteiro, E., "**Middleware Group Communication Mechanisms in M2M environments**", in the 2nd Joint ERCIM eMobility and MobiSense Workshop, 2013.
- Riker, A. and Cruz, T. and Marques, B. and Curado, M. and Simoes, P. and Monteiro, E., "**Efficient and secure M2M communications for smart metering**", in the Emerging Technology and Factory Automation (ETFA), IEEE, 2014.
- Riker, A. and Cerqueira, E. and Curado, M. and Monteiro, E., "**Data aggregation for group communication in machine-to-machine environments**", in the Wireless Days (WD), IFIP, 2014.
- Riker, A. and Cerqueira, E. and Curado, M. and Monteiro, E., "**Data aggregation for machine-to-machine communication with energy harvesting**", in the International Workshop on Measurements and Networking (M&N), IEEE, 2015.
- Riker, A. and Cerqueira, E. and Curado, M. and Monteiro, E., "**Controlling Data Aggregation and communication periodicity in Energy Harvesting networks**", in the IEEE Online Conference on Green Communications (OnlineGreenComm), IEEE, 2015.
- Riker, A. and Curado, M. and Monteiro, E., "**Neutral Operation of the Minimum Energy Node in energy-harvesting environments**", in the IEEE Symposium on Computers and Communications (ISCC), IEEE, 2017.

Co-advisor of MSc thesis:

- Subtil, J. and Riker, A. and Curado, M., "**Efficient Communication in Dense Networks**", University of Coimbra, 2017.

Special session presentations:

- Riker, A. and Curado, M. and Monteiro, E., "**Gestão de Grupos e Comunicação eficiente em M2M**". 8º Congresso Luso-Moçambicano de Engenharia.

Contents

List of Figures

List of Algorithms

List of Tables

Abbreviations and Acronyms

3GPP	Third Generation Partnership Project
ACK	Acknowledgement
ADSL	Asymmetric Digital Subscriber Line
CSMA	Carrier Sense Multiple Access
CPU	Central Unit Processing
CCA	Channel Clear Assessment
CON	Confirmable
CoAP	Constrained Application Protocol
CoRE	Constrained RESTful EnviRonments
DAV-NET	Data Aggregation for energy harVesting NETworks
DAMiG	Data Aggregation for Multiple Groups
DCT	Data Capture Task
DAO	Destination Advertisement Object
DODAG	Destination Oriented Directed Acyclic Graph
DAG	Directed Acyclic Graph
DIO	DODAG Information Object
DIS	DODAG Information Solicitation
EESR	Energy Efficient Spanning tRee
XML	Extensible Markup Language
ExTree	External Group Aggregation Tree
ExPath	EXternal group Path
EGReceiver	External Group Receiver
GTP-U	GPRS Tunneling Protocol User
GRASS	Grid-based Routing and Aggregator Selection Protocol
GM	Group Manager
HHA	Heterogeneous Hierarchical Architecture
HSC	High State-of-Charge

HEED	Hybrid Energy-Efficient Distributed clustering approach
M2M	Human-to-Human
HTTP	Hypertext Transfer Protocol
ILP	Integer Linear Program
ISC	Intermediate State-of-Charge
IntraTree	Internal Group Aggregation Tree
IGSink	Internal Group Sink
IETF	Internet Engineering Task Force
6LoWPAN	IPv6 over Low-Power Wireless Personal Area Networks
RPL	IPv6 Routing Protocol for Low-Power and Lossy Networks
KiBaM	Kinetic Battery Model
LP	Linear Programming
LA	Local Aggregators
L-PEDAP	Localized Power-Efficient Data Aggregation Protocol
LTE	Long Term Evolution
LEACH	Low Energy Adaptive Clustering Hierarchy
LSC	Low State-of-Charge
M2M	Machine to Machine
MA	Master Aggregators
MAT	Message and Actuation Tasks
MQTT	Message Queuing Telemetry Transport
MULTIPLEx	Messaging mULTIple Payloads LayEr
MCU	MicroController Unit
MST	Minimum Spanning Tree
MME	Mobility Management Entity
NST	Network Spanning Tree
NOMEN	Neutral Operation of the Minimum Energy Node
NiCd	Nickel-Cadmium
NiMH	Nickel-Metal Hydride
NON	Non-confirmable
NP	Non-deterministic Polynomial

PEDAP-PA	Power Efficient Data gathering and Aggregation Protocol-Power Aware
PEGASIS	Power-Efficient GAthering in Sensor Information Systems
PDT	Processing and Decision Task
RAM	Random Access Memory
ROM	Read-Only Memory
REST	Representational State Transfer
RST	Reset
RAME	Routing and Aggregation for Minimum Energy
ROLL	Routing in Low Power and Lossy networks
STT	Set of TTAMA Trees
SPT	Shortest Path Tree
SCCD	Single Cycle of Charge and Duty
SoC	State-of-Charge
TAUs	Tracking Area Updates
TTA	Two Tier Aggregation
TTAMA	Two-Tier Aggregation for Multi-target Applications
URI	Uniform Resource Identifier
UDP	User Datagram Protocol
WCDS	Weighted Connected Dominating Set
WiMAX	Worldwide Interoperability for Microwave Access

Chapter 1

Introduction

> It's the job that's never started as takes longest to finish.
>
> *(J.R.R. Tolkien, The lord of the rings)*

Contents

This thesis aims to improve the energy efficiency of machine-to-machine group communication. In order to achieve that, a comprehensive literature review was conducted to identify relevant open issues on the current research. To fulfill the identified gaps in the current approaches, a considerable number of mechanisms were proposed and evaluated using simulation, mathematical models, and real experiments.

Next, the overall research motivations of this thesis are presented followed by the detailing of the proposed objectives and contributions, as well as the thesis outline.

1.1 Background and Motivation

In the last years, all the electronic equipment keeps getting smaller, more powerful and less expensive. This trend towards miniaturization is a process that has been continuing for decades. The semiconductor industry has not only led to the miniaturization of transistors but also other areas of electronics, including sensors, transducers, actuators, communication modules, satellite transceivers and antennas. Now, several very small components are available on the market with an affordable price due to the large demand for these products. Running in parallel with this trend, communication networks have provided a huge range of technologies to interconnect the devices. There are technologies for all types of distances, energy consumption, mobility and bandwidth. Special attention has been placed on wireless technologies, since they ensure ease of deployment, the mobility of the devices and reduced costs.

This context of cheaper, more powerful devices and low-cost wireless network connectivity drove to the concept of Machine-to-Machine (M2M). M2M can be defined as the technology that is able to provide remote control for a large number of devices with little or no human intervention [Zhang et al., 2011b].

The market regards M2M communication as a key technology that is able to reach a huge new market, since the traditional Human-to-Human (H2H) market is showing signs of saturation. While the mobile data rates are predicted to double annually, the average H2H market revenue per user is likely to remain fixed [Marsch et al., 2012]. To date, there are about 8 billion mobile devices, including mobile phones and M2M devices [Cisco, 2016]. This number is expected to grow, since the number of connected devices in the world is expected to be 20 billion in the end of 2020 [Xia et al., 2017]. In the light of this, there is a great expectation that M2M will create a new market for device manufacturers, application developers and mobile network operators.

With the aid of M2M technology, transport systems (e.g. railways, highways and bridges) can be monitored with a large number of devices, since deploying multiple sensors in a monitoring system is no longer cost prohibitive. Moreover, implanting devices equipped with multiple sensors and actuators has become much more practical, in view of the low costs and reduced sensor sizes. Having these characteristics, the M2M data traffic is composed of multiple types with heterogeneous requirements [Taleb and Kunz, 2012].

Machine-to-Machine does not only refer to a large number of devices, it also involves numerous applications and data-consumer entities. Using a service platform (middleware) multiple applications and entities are able to request data from M2M devices. The middleware layer promotes the necessary interoperability between many stakeholders, and it reduces the costs with software development [Zhang et al., 2010]. With the aid of middleware service layer, multiple M2M data-consumers can request data from the M2M devices, concurrently. The responses to these data requests are frequently produced by groups of M2M devices that satisfy the data-consumers criteria (e.g. geographic location and data-type). So, it is necessary to perform and manage concurrent data gathering involving groups of devices in machine-to-machine systems.

The success of M2M systems can only be assured by having mechanisms that are able to improve the energy efficiency of M2M group communications. To achieve this, special attention has been given to the transceiver, as it is a component that most consume energy. It is a consensus that to improve energy efficiency, the actions (sniff, idle, receive and transmit) of the device's radio must be kept to a minimum and the sleep or the low power state must be prolonged. A manner to use the transceiver at minimum is to regulate the amount of traffic communicated by the group of nodes.

Under the aforementioned aspects, M2M communication has to employ mechanisms to regulate the amount of data transmitted. In this direction, data aggregation approaches have emerged as important mechanisms for group communication. It is able to summarize the data traffic, reduce the volume of data communicated, and minimize the communication overhead. In general, data aggregation is based on the premise that a set of nodes produce correlated data [Fasolo et al., 2007]. This premise is true if there is a group of nodes sensing the same particular environment or phenomenon, which is the case of group communication in M2M scenarios.

In view of the fact that data gathered from the group has some level of redundancy, data aggregation can extract the neighboring-data along the route and apply a summary function to reduce the amount of data [Fasolo et al., 2007]. Routing functionalities are needed to execute the data aggregation over the network, since routing influences the points of aggregation and determines the network flows. In addition to the points of aggregation, data aggregation mechanisms must coordinate the amount of data on transmissions, by defining an aggregation level. Otherwise, congestion or inefficient use of energy may occur.

A number of different solutions have been proposed to address energy efficiency in general M2M communication. Several of these proposals are reviewed and summarised in Chapter 3. However, the literature review shows that there are important open issues to be solved specifically in M2M group communication. To fill these gaps, this thesis proposes, discusses and evaluates group communication mechanisms able to improve the energy efficiency by regulating the network traffic and reducing the communication overhead.

1.2 Objectives and Contributions

The main goal of this thesis is to improve energy efficiency of machine-to-machine group communication. As the definition of energy efficiency is broad, special attention should be given for traditional battery-powered with limited energy resources and renewable scenarios. In battery-powered scenarios, the nodes have a fixed amount of energy, and the network lifetime is the main performance metric to be maximized. In the other case, in renewable scenarios, nodes can recharge their reserves, so their spent of energy must be controlled to achieve sustainable consumption, which means to avoid battery depletion or overflow.

This thesis contains an in depth literature review related to machine-to-machine group communication and energy efficiency. Besides, it follows a research direction able to mitigate relevant open issues in this field. The proposed mechanisms for machine-to-machine group communication were implemented and evaluated using distinct methodologies, such as simulation, mathematical models, and testbed experimentation.

The specific goals of this thesis are as follows:

Goal 1 - Define a generic approach suitable and adaptable for machine-to-machine group communication to eliminate data redundancy and reduce overhead;

Goal 2 - Propose a set of group communication mechanisms that aims to improve the energy efficiency in traditional battery-powered devices with fixed energy budget and renewable energy sources;

Goal 3 - Study the performance of the proposed mechanisms.

Taking into consideration the specific goals, this thesis has produced the following contributions:

Contribution 1, A generic approach for M2M group communication

This contribution is related to the proposal of a generic approach suitable to improve M2M group communication. The proposed approach applies two distinct types of processing functions on the group traffic to enable less overhead and eliminate data redundancy. This approach, Two Tier Aggregation, is flexible enough to be used on different group communication scenarios and allows the groups to have particular communication settings. All the mechanisms and models presented in Chapter 4 benefit from this contribution, which is shown in Section 4.2;

Contribution 2, Heuristics to prolong the network lifetime

Routing heuristics play an essential role in the performance of the proposed Two Tier Aggregation. The heuristic decides aspects related to the points of aggregation and how traffic will flow over the network. The proposed heuristics are tree-based solutions that find compatible groups to reduce the overhead and eliminate the data redundancy. This contribution is presented in Section 4.3;

Contribution 3, Model formulation for network lifetime maximization

The formulation of mathematical models is a powerful technique able to find the optimal solution for a given problem. Even though the computation cost to find the optimal solution is too high to be implemented, the optimal solution is important for the study of a problem because it defines an upper bound. In this case the network lifetime upper bound was obtained using Integer Linear Programming and allowed the comparison between the optimal and the heuristics solutions. This contribution is introduced in Section 4.4;

Contribution 4, Prototyping and testbed experimentation

This contribution is related to the implementation and evaluation of the main proposed functionalities in real devices. The experimentation using prototypes and real environments poses several challenges not commonly considered in simulation or mathematical models. Using real devices, it is possible to have a realistic notion of the gains and limitations of the proposed procedures. The conducted testbed experimentation was focused on measuring the energy consumption, testing basic cases and analyzing the impact the aggregation procedures have on the energy. This contribution is detailed in Section 4.5;

Contribution 5, Mechanisms for energy sustainable operation

Mechanisms were proposed to keep in sustainable operation devices with energy recharging sources. The focus of these mechanisms was to regulate the amount of traffic flowing through a node using dynamic data aggregation procedures. In the first proposed mechanism, each node regulates autonomously its traffic using as main parameter the residual energy in the reserve. In the second mechanism, each node decides the amount of traffic by knowing the minimum energy level in the path. This contribution is presented in Sections 5.2 and 5.3;

Contribution 6, Adaptation and extension of standard M2M protocol stack

There are several systems and platforms to facilitate implementation and support the solution embedding on real devices. However, some of these systems implement their own protocol stack or do not give access to the source code. Instead of using proprietary set of protocols, the standard M2M protocol stack was adapted and extended to promote the convergence and the advance of the M2M protocols in energy harvesting scenarios. This contribution is presented in Section 5.3.3.

The next section shows the outline of the thesis.

1.3 Thesis Outline

The reminder of this thesis is organized in five chapters, as described below.

Chapter 2 - Machine-to-Machine Environments

This chapter introduces the main concepts and covers a set of background notions necessary for the understanding of the literature review and the proposed solutions. The emphasis of this chapter is to study machine-to-machine applications, the different types of communication, and the network architecture. This chapter also presents technologies related to group communication in M2M environments, such as the Constrained Application Protocol and the IPv6 Routing Protocol for Low-Power and Lossy Networks. This chapter also details the general challenges and also presents specific machine-to-machine scenarios. The specific scenarios are important to narrow the scope of the research and were used as base to identify and list a set of communication requirements;

Chapter 3 - Related Work

This chapter presents a state-of-the-art study conducted to analyze the current solutions most suitable to fulfill the listed communication requirements. The final outcome of this chapter is the identification of open research issues, which serves as research direction for the design of the proposed solutions;

Chapter 4 - Mechanisms for Energy Efficiency in M2M Group Communication

This chapter proposes a generic approach, called Two Tier Aggregation, to improve the energy efficiency of machine-to-machine group communication. In addition, this chapter proposes and assesses two mechanisms that apply Two Tier Aggregation on heterogeneous group communication applications. The first mechanism, named Data Aggregation for Multiple Groups (DAMiG), computes independent trees to communicate and aggregate the internal and the external group traffic. The second mechanism, named Two-Tier Aggregation for Multi-target Applications (TTAMA), computes a spanning tree including all active groups and then prunes the unnecessary nodes. This chapter also shows the formulation of a mathematical model able to find the maximum network lifetime of the Two Tier Aggregation approach. A performance comparison is made between the proposed mechanisms and optimal and near-optimal solutions. Protopying and testbed experiments are also described in this chapter, including the measurements aiming to find the effective amount of energy the proposed aggregation procedures saves;

Chapter 5 - Dynamic Data Aggregation in Replenishable Energy Environments

This chapter proposes and evaluates two mechanisms that use data aggregation procedures to regulate the energy consumption, seeking to maintain the nodes in neutral operation. The first mechanism, Data Aggregation for energy harVesting NETworks (DAV-NET), uses the residual energy on the battery to define an aggregation level. The second mechanism, Routing and Aggregation for Minimum Energy (RAME), follows a similar idea, but it uses the minimum residual energy in the path to determine the amount of data to be aggregated. In comparison to DAV-NET, RAME is a much more comprehensive solution since it focus on keeping at neutral operation the nodes with minimum energy reserves. Besides, it adapted and extended the standard protocols Constrained Application Protocol (CoAP) and IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL);

Chapter 6 - Conclusion and Future Works

This chapter presents the final remarks and conclusions taken from this thesis, as well as the research directions for future works.

The next chapter presents the fundamentals and background concepts related to machine-to-machine communication.

Chapter 2

Machine-to-Machine Environments

All my movements are carefully
premeditated.

(El Chapulín Colorado)

Contents

Machine-to-Machine communications have presented a large growth on several fields and have driven the emergence of a new era of smart applications. This chapter explores important topics of machine-to-machine environments, including the aspects related to scenarios, applications, network architecture, data traffic, and technologies.

2.1 Introduction

Machine-to-machine systems have brought great potential opportunities for designing monitoring and remote control applications. It is seen as a key technology to connect the huge number of embedded devices, which is increasing tremendously, and to accelerate the development of numerous innovative applications. Most part of the academic community agrees with the following broad definition for machine-to-machine: “M2M refers to a system formed by a large number of low-power and low-cost embedded devices sharing information and making collaborative decisions with little or no human intervention” [Zhao et al., 2017].

For an understanding of machine-to-machine environments, Figure 2.1 shows the continuous workflow performed by a generic M2M system. The first task that must be performed by the M2M devices is Data Capture Task (DCT), which involves collecting data from the real world. Examples of electronic sensors that carry out this task are: temperature, humidity and flow measurement.

After the DCT, the machines must perform the Processing and Decision Task (PDT). This task requires powerful computational capabilities to handle the data acquired, and support decision-making functionalities.

M2M systems have devices able to depict messages and execute actions. This task is called Message and Actuation Task (MAT) and its purpose is to provide the environment with system responses. These responses might be alerts, information, messages, commands or important events for actuators. The MAT is the output of the M2M system and its destination might be humans and/or machines.

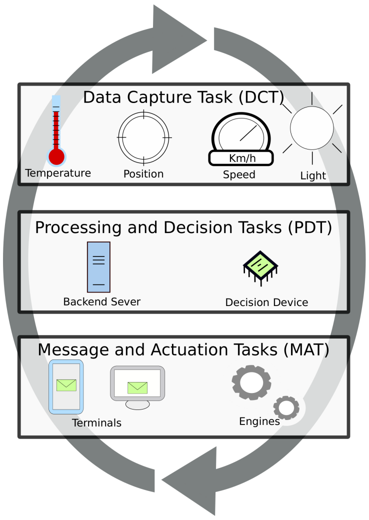


Figure 2.1: Basic M2M Workflow Tasks.

The M2M system becomes more complex when these three basic tasks are arranged in a distributed fashion. In the light of the distributed execution of these tasks, the following classes of machines emerge in the M2M system:

- **Pure DCT machines:** Devices only equipped with the necessary hardware/software to sense the environment.
- **Pure PDT machines:** Devices able to compute the data and choose the appropriate action based on a set of rules or policies.

- **Pure MAT machines:** Devices that have the resources necessary to perform some automated actions and/or display alerts/messages.
- **Hybrid MAT/DCT machines:** Devices able to act either as a sensor or actuator/informant.

Figure 2.2 depicts these distributed arrangements of functionalities and the set of machines with different capabilities. It should be noted that some of the PDT machines might have different levels of resources (e.g. computing power, memory and power supply) and might be deployed at distinct distances from the other machines.

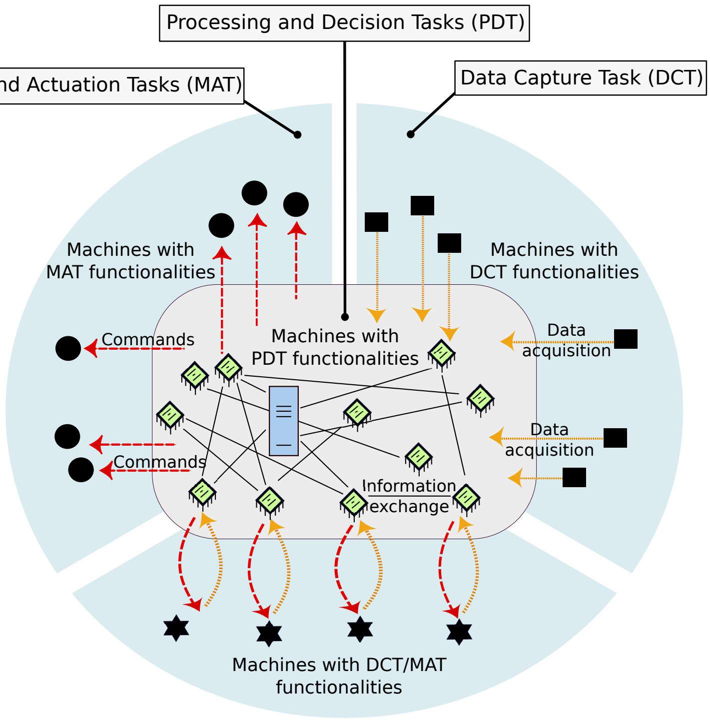


Figure 2.2: Types of devices in M2M environments.

In the current stages of M2M systems, the typical system architecture is composed of machines with MAT and/or DCT capabilities remotely connected to centralized PDT machines. However this architecture has been extended to scenarios involving machines with multiple functionalities (MAT/DCT) that communicate with PDT functionalities distributed over several devices [Mehmood et al., 2017].

Important notions to be extracted from the machine-to-machine systems are: (i) M2M systems are composed of devices with very distinct hardware and software capabilities; and (ii) M2M devices perform their functionalities in a distributed and complementary fashion.

The remainder of this chapter presents and further discusses the main concepts of machine-to-machine communication. Section 2.2 presents the scenarios of applications, the types of data traffic, and the network architecture. Section

2.3 details important standards defined for M2M communication. Section 2.4 introduces the main research challenges in M2M communications.

2.2 Machine-to-Machine Communication

After studying M2M systems from a high level perspective, it is important to describe and show the characteristics of some M2M applications, since the M2M systems have the potential to be applied to several scenarios.

2.2.1 Scenarios and Applications

Many companies and industries are still exploring M2M scenarios and applications. The descriptions of the main M2M scenarios and applications are provided below.

- **Smart Grid and Smart Metering:** The aim of smart metering is to measure the consumption of the utilities, control the assets, enhance the performance of the system and provide security. It is also designed to reduce operational, maintenance and system planning costs [Ghavimi and Chen, 2015]. Smart metering is also used to monitor and control drills, wells and pipelines. Some of these facilities are characterized as inaccessible, and M2M communication becomes a viable option to monitor the extraction, transport and distribution of oil, gas, electricity and water.

- **Automotive:** The automotive market has attracted many stakeholders (e.g. automotive consumers, manufacturers and providers of automotive services). Most of the manufacturers are involved in the creation of applications, for instance, safety and collision avoidance, traffic management infrastructure systems, vehicle telematics, in-car entertainment and internet services [Booysen et al., 2012]. In the M2M automotive scenarios, the driver is able to obtain information about traffic, accidents, natural disasters, climatic conditions, route assistance and Internet access. By having access to mechanical reports, the automobile manufacturers are able to make progress in the research and development of new technologies to further improve efficiency, increase driver satisfaction and provide environmental benefits.

- **Health:** Healthcare applications enables data collection about the body condition (e.g. blood pressure, body temperature, heart rate, weight and body locations) via body sensors. The basic idea of Healthcare applications is to forward the data gathered from the patients to clinical centers, where the physical indicators can be handled through remote monitoring. The analysis by the appropriate professionals allows the patients to be given better treatment or some feedback which is useful, for instance, for better fitness activities.

- **Smart Cities:** Currently, cities need to have a better infrastructure and resource management. This has emerged from the trend toward greater urbanization, as well as population growth and the need for environmental sustainability. City monitoring is a scenario, in which the M2M system encourages the integration of private and public services. The data collected by the M2M communication system could be used to plan the supply of new resources, manage current services (e.g. public transport information and traffic control) and implement services (e.g. police operations in strategic locations).

- **Monitoring and Control of Objects:** Keeping track of objects and/or animals is also an important field for M2M applications. A lot of companies and people are interested in keeping track of their possessions. The M2M system can provide a comprehensive view of the state of their assets so that they can take action when necessary. In many situations, companies are interested in carefully controlling their properties by means of remote monitoring. For instance, in a rural area, horses could be monitored (e.g. their position, health conditions and the temperature of the herd). In an urban area, the M2M monitoring applications could be used against burglary or for alerting a distributor when a vending machine is out of drinks or the maintenance staff when an air-conditioning unit has a malfunction.

Table 2.1 summarizes the major areas of M2M and gives some examples of specific applications.

Table 2.1: M2M Major Areas and Specific Applications (adapted from [Pandey et al., 2011]).

Major Areas	**Specific Applications**
Automotive	Failures calls, stolen vehicle tracking, remote diagnosis, insurance services, fleet management, emission control, toll payment, road safety and navigation
Smart Metering	Management of electricity distribution network, meter reading communication, management of power quality and outage, tariff setting and payment, theft protection
City monitoring	Traffic flow management, passenger information system, dynamic traffic light control, surveillance and security
Healthcare	Monitoring of patients' health and fitness, triggering alarms in critical conditions, remote control of medical treatment
Track and control (Person/Objects)	Stock monitoring, production line monitoring and theft protection, identifying faults

The next section presents the existing types of traffic in M2M communications.

2.2.2 Data Traffic

Before M2M, the usual Internet traffic was mostly created by people accessing information. The advent of M2M communication results in new types of network traffic. The M2M traffic has emerged in the Internet and there are expectations that there will be a much larger number of smart devices that need to be supported in the Internet than the traditional computers used at present [Beale and Morioka, 2011].

Traditional Internet services, such as, web-browsing, file downloading and webmail, are characterized by large blocks of data. On the other hand, M2M communication is characterized by having a wide variation of traffic characteristics [Lioumpas et al., 2011]. For instance, some M2M applications generate infrequent and delay-tolerant packets (e.g. smart metering and parking metering), while other M2M applications require real-time transmission (e.g. video surveillance). Also, there is periodic and event-based traffic.

On the basis of the Technical Report TS 22.386 drawn up by the 3GPP [(3GPP), 2010], the main characteristics of M2M traffic are listed as follows.

- **Time controlled:** M2M data delivery only during predefined time intervals.
- **Time tolerant:** This is the data which does not have delay requirements.
- **Infrequent transmission:** This refers to data that is hardly ever generated.
- **Small data transmissions:** It is the feature related to the applications that only communicate small amount of data.
- **Mobile originated:** This is the data generated in mobile situations.
- **Low mobility:** It is the communication generated by devices that rarely move or only move within a certain geographic area.
- **Events/entities:** Monitoring events related to particular M2M devices.
- **Priority alarm:** This is the communication caused by the occurrence of special events or situations.
- **Group Based Communication:** This is the communication involving a set of M2M devices.
- **Two-Way:** It is important to notice that M2M communication involves two-way (forward and backward) communication, and each of them have distinct latency requirements to be satisfied.
- **Multiple Data-Type:** In general, each M2M application communicates more than a single data-type. This is because the M2M devices are equipped with multiple sensors, and some of the traffic characteristics are related to mobility, frequency, amount of data, delay tolerance, priority and the number of devices involved in the communication.

The generation of a particular type of data traffic depends on the characteristics and implementation of the applications. For example, a heart monitoring application could be implemented to communicate the health reports at regular time intervals or when some emergency situation occurs. In the case of the former, the application requires continuous monitoring, which means the data traffic can be characterized as time controlled. In the case of the latter, the application only generates the notification when an emergency event is detected, so it produces priority alarm traffic.

The study of types of traffic produced by M2M devices is useful to understand the network architecture, which is the subject of the following section.

2.2.3 Network Architecture

The combination of wired and wireless network technologies may result in several variants of network architectures able to provide communication for the M2M systems. By integrating network technologies, some solutions ([Mehmood et al., 2017] and [Ghavimi and Chen, 2015]) have shown a Heterogeneous Hierarchical Architecture (HHA).

As can be observed in Figure 2.3, the lowest layer is the M2M Capillary Network, which is formed of M2M Devices and M2M Gateways [Ashrafuzzaman and Fapojuwo, 2018]. These devices can be connected via Bluetooth [Bluetooth, 2001], Zigbee [Alliance, 2006], or IEEE 802.15.4, which run on the 2.4 GHz band.

The M2M Devices are designed to maintain low cost per unit, allowing the deployment of a large number of units. In addition, the hardware capabilities of the M2M Devices are restricted in terms of Central Unit Processing (CPU), battery autonomy and memory. This limited hardware capacity imposes severe restrictions on the functionalities carried out by M2M Devices. In general, they have low transmission rates and perform simple operations (e.g. sensor or switches off/on). In contrast with M2M Devices, most M2M Gateways have a constant energy supply and robust hardware.

The M2M Access Network Layer is the communication system that provides Internet connection (Core Networks) to the M2M Gateways. Examples of M2M Access Network technologies are: Ethernet, Wi-Fi [Alliance, 2007], Asymmetric Digital Subscriber Line (ADSL) [Standard, 1998], Long Term Evolution (LTE) [Access, 2007] and Worldwide Interoperability for Microwave Access (WiMAX) [Group et al., 2004].

By means of M2M Gateways, the Hierarchical Heterogeneous Architecture reduces the number of machines that are directly connected, as well as the traffic load and the number of subscribers over the Access Network Layer. M2M Gateway nodes must forward the data from M2M Devices to the Access Network Layer and vice-versa, by supporting connections between both layers [Naeem et al., 2017]. For instance, when an external node needs to contact a M2M Device that is attached to a M2M Gateway, this external node must first contact the appropriate M2M Gateway, and then, the M2M Gateway is able to

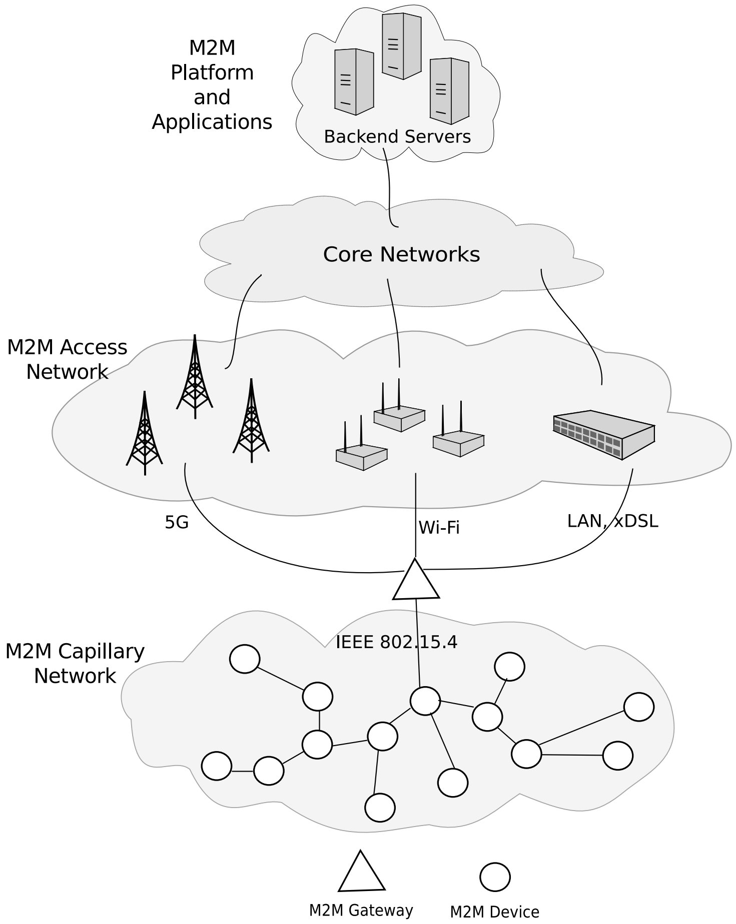


Figure 2.3: Network Architecture of M2M Communication.

establish a connection with the particular M2M Device. This means that the M2M Gateway can act as a proxy, by providing a service of remote access [Ishaq et al., 2016].

The M2M Platform and Application layers are at the top of the HHA. These layers are the destination of the information collected by the nodes. The M2M Platform and Application Layer are usually hosted in the Backend Servers and equipped with powerful hardware resources that are able to process a large number of transactions [Matoba et al., 2011].

The M2M platform (also called middleware) can be defined as a software layer that enables M2M applications to access their devices by using a common set of services. The wide range of devices in terms of capabilities, software complexity and costs, and the necessary interoperability between many stakeholders, have resulted in the adoption of a M2M middleware [Zhang et al., 2010]. This adoption enables the interaction of different applications from different stakeholders and reduces the programming costs.

The next section details some standards defined for machine-to-machine communications that will be mentioned in the remaining chapters of this thesis.

2.3 Standards Related to M2M Communication

The Internet Engineering Task Force (IETF) is one of the leading institutions aiming to define standards and protocols for machine-to-machine communication. IETF has created working groups to address different aspects in this field. One working group is the Routing in Low Power and Lossy networks (ROLL), which addresses all the routing aspects. Other working group, called Constrained RESTful EnviRonments (CoRE), was established to make progress on a framework that enables embedded web for constrained M2M devices.

The next section details the Constrained Application Protocol (CoAP), which is the resulting protocol from the CORE working group.

2.3.1 Constrained Application Protocol

Many internet applications depend on the Web architecture and use the Hypertext Transfer Protocol (HTTP) [Fielding et al., 1999]. Due to the success of HTTP, the Constrained Application Protocol (CoAP) [Shelby et al., 2014] is a lightweight application protocol inspired on the HTTP, but designed for M2M devices with low energy and computation resources. CoAP is based on the Representational State Transfer (REST) architecture, which identifies the information resources by Uniform Resource Identifier (URI).

The interaction model used by CoAP is based on the client and server paradigm, where a client requests an action on a resource located at the server. The CoAP messages can be Confirmable (CON), Non-confirmable (NON), Acknowledgement (ACK) and Reset (RST).

Confirmable messages implement a retransmission mechanism using a default timeout and exponential back-off until an Acknowledgement message is received. The confirmable message are implemented because CoAP assumes the unreliability of the transport layer, for instance User Datagram Protocol (UDP). Additionally, the option of Non-Confirmable is also provided for applications that do not require reliable communication.

Each CoAP message carries a Message ID and Token in the header, as depicts Figure 2.4. The Message ID is a 16-bit code that is used to detect duplicated messages, while the token is a sequence of 0 to 8 bytes that is used to match a response with a request. Every request carries a client-generated token that the server must echo (without modification).

Another field in the CoAP header is the Code. It is a 8-bit unsigned integer, in which the first 3-bits are used to identify the class and the remaining 5-bits determine the details. The classes defined so far are: request (0), a success response (2), a client error response (4), and a server error response (5).

The main code values in request and response messages are presented in Table 2.2.

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |Ver| T |  TKL  |      Code     |          Message ID           |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   Token (if any, TKL bytes) ...
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   Options (if any) ...
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |1 1 1 1 1 1 1 1|    Payload (if any) ...
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2.4: CoAP message fields.

Table 2.2: Codes in CoAP messages.

Type	**Code**	**Name**
Request	0.01	GET
	0.02	POST
	0.03	PUT
	0.04	DELETE
Response	2.01	Created
	2.02	Deleted
	2.03	Valid
	2.04	Changed
	2.05	Content
	4.00	Bad Request
	4.01	Unauthorized
	4.02	Bad Option
	4.03	Forbidden
	4.04	Not Found
	4.05	Method Not Allowed
	4.06	Not Acceptable
	4.12	Precondition Failed
	4.13	Request Entity Too Large
	4.15	Unsupported Content-Format
	5.00	Internal Server Error
	5.01	Not Implemented
	5.02	Bad Gateway
	5.03	Service Unavailable
	5.04	Gateway Timeout
	5.05	Proxying Not Supported

Next, the Observe and Group-Communication options of a CoAP request message are detailed.

2.3.1.1 Observe

The Observe option of CoAP request gives the capability to receive periodically data updates of a data resource located in a CoAP server. Using this option, CoAP clients do not send a CoAP GET request frequently for the same data resource. Instead, with a single Observe GET request, the client subscribes for a particular data resource.

The Observe option is important for M2M communication, since it avoids repeated requests for the same data resource. Besides, it allows the execution of an asynchronous communication model, since the server updates the client every time the Observed data has filled a criteria.

Next, other important CoAP option, group communication, is described.

2.3.1.2 Group Communication

A particular scenario for CoAP is when a client wants to communicate to a group of server nodes. In this case, information must be sent to or received from a group of devices, allowing a single CoAP client to get or send simultaneously data resources from or to multiple CoAP servers.

CoAP is designed to support group communication to improve efficiency. A CoAP group is defined as a set of CoAP endpoints, where an endpoint is an entity participating in the communication as client or server.

A CoAP server is allowed to belong to multiple groups and group membership of a node may dynamically change over time. Originally, the CoAP standard defines that the CoAP groups can be reached through group's associated IP multicast address. However, the individual response by each group member to a CoAP group communication request is always sent back as unicast.

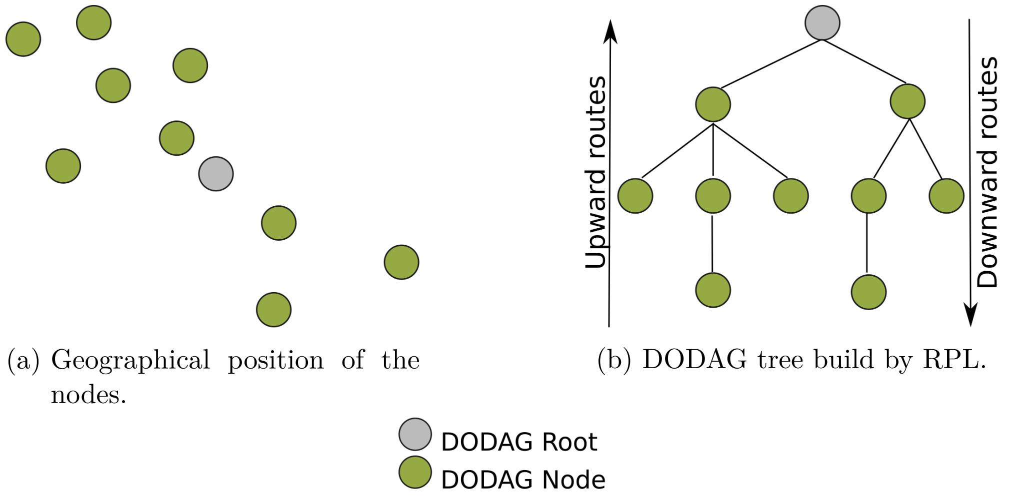
The next section presents another protocol produced by the IETF, the RPL protocol.

2.3.2 Routing Protocol for Low-Power and Lossy Networks

The reduced energy and hardware resources of M2M devices creates a network characterized by high loss rates, low range, low data rates, and link instability. In this context, the Internet Engineering Task Force has maintained the Routing Over Low power and Lossy networks (ROLL) working group since 2008. To meet the requirements of the Low Power and Lossy Networks ROLL has developed the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL).

The special focus of RPL is on multipoint-to-point communication where multiple devices send traffic toward a central point, and vice-versa (i.e. point-to-multipoint). The central point is also called sink node and most of the times it acts as Gateway. RPL creates and maintains a tree topology, named Destination Oriented Directed Acyclic Graph (DODAG), rooted at the Gateway. Multiple instances of RPL can coexist in the network and multiple DODAGs can be constructed in each instance, but a node belongs to a single DODAG for each RPL instance it joined.

The idea of multiple RPL instances comes from the necessity of optimizing the network paths for different objectives. The same network nodes can run



(a) Geographical position of the nodes.

(b) DODAG tree build by RPL.

Figure 2.5: RPL DODAG tree.

simultaneously two or more RPL instances to serve a performance criteria, given by the Objective Function. For instance, one RPL instance seeks to reduce the latency while another aims to reduce the energy consumption.

In RPL, the main mechanism to detect and avoid loops in the routes is Rank. It is a scalar representation of the node in the DODAG. Although the exact calculation of the Rank is specified by the Objective Function, the general principle of Rank is a metric that reveals the node's individual position relative to other nodes taking the root as reference. The Rank value must monotonically decrease as a route is made in the DODAG tree towards the root. In the other case, the Rank must monotonically increase going from the root to the nodes.

As shown in Figure 2.5, RPL distinguishes Upward and Downward traffic according to the direction the messages travels. Being more specifically, Downward is the term used to indicate that the communication is going from the root to the nodes, and Upward indicates that information is flowing from the nodes to the root.

The DODAG tree is created in a distributed way. The process initiates with the announcement of a new DODAG, to which the nodes can join. When a new node join the DODAG, it must provide at least one parent node as a default route towards the root. A parent of a node is defined as one of the immediate successors of the node on a path towards the root. A node that has not all the resources necessary to act as parent may join the DODAG as leaf node. To find the parent towards the root and also to enable the communication between the root to the nodes, RPL exchanges frequently three types of control messages:

- **DODAG Information Object (DIO):** Message that contains information that makes a RPL Instance discoverable. This message carries the RPL instance ID, DODAG ID, Mode of Operation, Rank of the sending node, Metric Container, Routing Information, and DODAG Configuration.

- **DODAG Information Solicitation (DIS):** This message has the simple objective to solicit a DIO from a RPL node. A node also may use DIS to probe its neighborhood for nearby DODAGs.
- **Destination Advertisement Object (DAO):** This message is used to propagate destination information Upward along the DODAG. It means that this message is used to support Downward routes.

The only manner a node can join to a DODAG tree is using information contained in DIO messages. Due to this reason DIO messages carry more information than other RPL messages. The DAO message also propagates RPL routing information, but it contains Upward route information which is valid only to members of the same DODAG.

RPL defines some modes of operation, which are related to the type of support it gives to the Downward routes: (i) Non-storing mode: No Downward routes are maintained by the regular nodes, which means that the information flows from the nodes to the root and only the root stores the routing table entries for destinations found by DAO messages; and (ii) Storing mode: all nodes, except root and leaves, store routing table entries for destinations learned from DAO.

Regarding the RPL mode of operation, when a node joins a DODAG, it must accept the operation mode defined by the root. In case a node does not support the selected operation mode, it joins the DODAG as a leaf node, which means that it cannot be parent of any other node in the DODAG tree.

The next section presents the challenges in Machine-to-Machine communication.

2.4 Challenges in Machine-to-Machine Communication

Several reports predict a huge growth for the M2M market in terms of devices and connectivity. It is estimated that the number of smart-metering devices per cell in a typical city will be tens of thousands [Akpakwu et al., 2018]. However, with this potential introduces additional obstacles. Thus, the success of M2M communication depends on the development of approaches able to tackle major challenges in business and in academic research.

One important business barrier for M2M success is the adoption of standard protocols and technologies. Without the insertion of standards in the market, different stakeholders cannot interplay easily. Low monetary cost per device unit is another business demand for M2M environments. The affordable price of the devices is essential to deploy hundreds of M2M devices on large areas.

Research challenges in machine-to-machine aim to improve the performance by providing new approaches to increase the energy efficiency, or provide more manageability or to support scalability of the network. Given the vast range of M2M applications, there are several research challenges. Some of them are relevant for this thesis, such as: energy efficiency, group communication, and traffic congestion in heterogeneous dense networks.

The next section gives an overview of the most relevant challenges related to this thesis.

2.4.1 Relevant Challenges

With regard to research challenges in M2M communication, many aspects must be investigated. The most relevant research challenges are shown in the next sections.

2.4.1.1 Energy Efficiency

The amount of energy resource available for the nodes is a factor that impacts the restrictions of M2M devices. As the energy is resource essential for all hardware components, M2M devices powered by small batteries must not be equipped with high consumption circuits, even though its cost may be low. Thus, energy consumption has effects on diverse system aspects, such as communication reliability, rate of data exchange, circuits design, and other hardware constraints.

To efficiently use the energy resource is a challenge in M2M communication, but the energy efficiency can admit different meanings depending on the energy sources the nodes have. In general, the nodes can have limited or replenishable energy resources. These cases are detailed next.

Battery-Powered
Typically, the M2M devices acting as sensor or actuators are powered by small batteries. A small energy reserve contributes for the low-cost of the devices and makes the deployment of a large number of nodes easier. In these scenarios, the energy efficiency challenge is tackled when the devices save the energy resources as much as possible. Or, in case of connected devices communicating cooperatively (e.g. relay and forward nodes), energy efficiency is achieved maximizing the network lifetime, which is a measurement related to the full operational time of the network [Azari and Miao, 2017].

Energy Harvesting
Some M2M devices have manners to collect energy from the ambient (e.g. solar, wind, acoustic, and vibration [Sudevalayam and Kulkarni, 2011]), which means that their energy resources are replenishable. So, not all M2M devices must minimize the energy consumption at all time. For these devices, energy efficiency means to use the amount of available energy at a rate that avoids energy waste and depletion [Ozel et al., 2015]. Theoretically, at an optimal consumption rate, the device can maintain its operation indefinitely, assuming the ambient can provide energy to be harvested [He et al., 2015]. This optimal operation is also

called neutral operation or self-sustainable operation and guarantees that the remaining energy on the battery is maintained at a desirable interval.

Next, the group communication challenge is detailed.

2.4.1.2 Group Communication

Most M2M applications, such as smart-metering and city-monitoring, involve a large number of devices. These devices, when connected, form a network that must be manageable. In practice, the management of M2M device involves firmware updates, anomaly detection, and storage of hardware information and reachable addresses. To reduce the complexity of the management operation, many of the management tasks must be executed in groups of devices [Ghavimi and Chen, 2015].

Groups or clusters can be formed according to the similarity of the devices or data. The possibilities for grouping the devices are vast. One of the examples is to group the devices that have the same class of sensors located in a given area. The existing groups allow a client to request data without specifying one-by-one the addresses of the target devices. Thus, the access for the group becomes transparent for any client interested on reaching the group.

Continuing the presentations of the challenges, the next section shows the one related to traffic congestion.

2.4.1.3 Heterogeneous Dense Networks

The straightforward approach to provide connectivity for M2M devices is using small cells of cellular networks [Ali et al., 2015]. From the network point of view, this solution is homogeneous, but it is cost-inefficient and it is not suitable for dense networks. This idea would result in heavy traffic signaling for controlling and managing dense areas of M2M devices, contributing to the network congestion.

Heterogeneous wireless connectivity is envisioned for M2M communication because it can be applied to reduce the number of machines connected to the cellular network or other technology. According to this concept, M2M devices are connected locally using a short or medium range wireless technology, and some special nodes, called gateways or border-routers, provide external connectivity. This challenge is to perform cooperative approaches on the network traffic to reduce the data reaching the gateway. Besides, the cooperative approaches, executed on the M2M capillary network, should avoid the overhead caused by several message transmissions containing small size payloads [Boccardi et al., 2014].

The next section discusses specifics scenarios and communication requirements.

2.4.2 Scenarios and Requirements

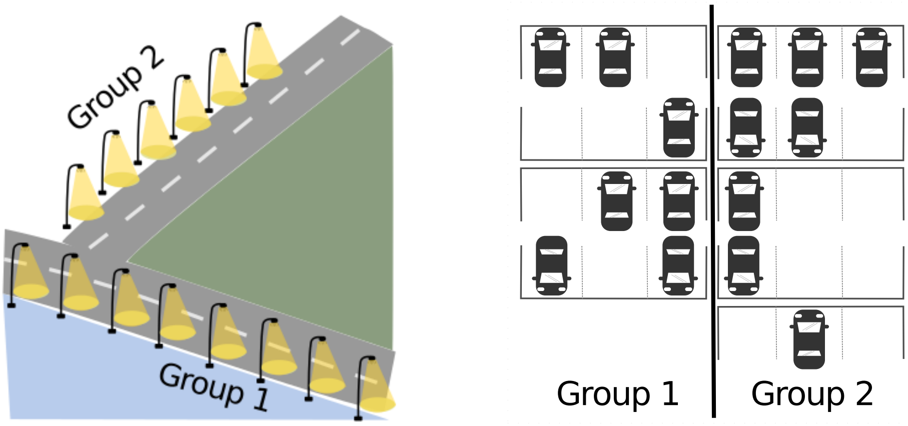
M2M communication involves diverse types of applications and its general challenges are very broad. To facilitate the requirements analysis of M2M communication, the following sections narrow the scenarios in analysis. Since efficient energy is transversal for the others challenges, each specific scenario considers different energy sources. In the first scenario, the nodes are powered by batteries, while in the second the devices are able to harvest energy.

2.4.2.1 Group Communication over Low Power Networks

A class of M2M scenarios is composed by the applications which clients need to monitor multiple groups [Teklemariam et al., 2013] and demands periodical communication from each group [Fadlullah et al., 2011]. Some CoAP options, observe and group communication, are in accordance with these scenarios. Next, some practical applications are presented.

Smart Parking and City Monitoring
In the City Monitoring application, see Figure 2.6.a, there are two groups of devices that can communicate different measurements, such as the levels of carbon dioxide emissions and electricity consumption. Each group communicates and produces a message revealing the measurements every 60 seconds.

Figure 2.6.b illustrates two periodic applications that monitor multiple groups over the network. In the case of the Smart Parking application [Zanella et al., 2014], the devices are divided into two groups and are responsible for giving information about the total number of available parking lots every 30 seconds.



(a) City Monitoring.
Sensors: Luminosity and Electric Current.
Update interval 60 seconds.

(b) Smart-parking.
Sensors: Temperature, and Presence.
Update interval 30 seconds.

Figure 2.6: Multiple Group Monitoring.

Building Automation
Another example of group communication, a M2M building automation applic-

ation must keep track of different geographic areas of the building so that it can make the appropriate operational decisions. In each building area, a group of devices must sense a particular target, such as temperature, humidity, light, and concentration of toxic gases. In this case, the geographic area and the type of data can be the criteria to define the targets that must be monitored (i.e. target 1="room temperature 1", target 2="room humidity 1", and target 3="the light in the room 2").

Based on these applications and on the aforementioned M2M challenges, it is possible to identify a set of requirements for M2M communication. These aspects are presented as follows.

1. **Eliminate data redundancy:** As a group of nodes monitors the same target, the levels of data redundancy are likely to be high [Al-Karaki et al., 2009]. Owing to limited resources, data redundancy should be kept to a minimum to avoid high communication costs and consumption of energy resources.
2. **Minimize overhead:** The energy consumed by the control messages should be kept to a minimum [Lo et al., 2013], since the message payload is small and frequent in group monitoring [Shafiq et al., 2012]. In practical situations, the payload content represents only ten percent of the total message size, but it can cause network congestion [Boccardi et al., 2014].
3. **Support heterogeneous groups:** Depending on the requirements of the clients involved in the group communication, the system should allow each group to execute its own configuration [Katsaros et al., 2014]. It means that each group can be configured with an unique setting. The groups might communicate with a particular periodicity (e.g. every 30 seconds), maximize a particular resource (e.g. energy), provide a tolerable level of performance (e.g. maximum delay and loss), and execute some in-network data processing functions.

The next section shows M2M scenarios where the nodes have the capability of harvesting energy from the ambient.

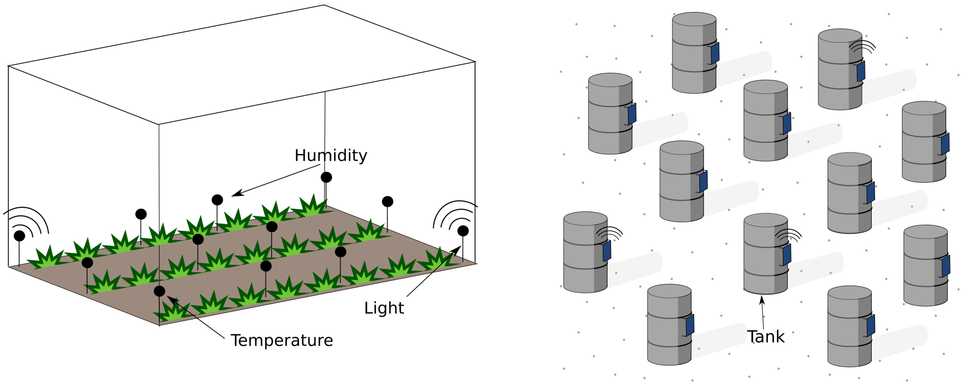
2.4.2.2 Energy Harvesting Networks

Another specific area of M2M scenarios is formed by applications that collect data from nodes able to harvest energy from the environment. Next, two energy harvesting scenarios are provided.

Smart Greenhouse and Tank monitoring
Figure 2.7.a shows applications where devices measure the luminosity, humidity, pressure and temperature. In the smart greenhouse application, the data collected from the nodes can be used to control the windows, lights, heaters, or fans. Besides, the data provided by the nodes is also an important record to study the growth of the species.

In the Oil tank monitoring application depicted in Figure 2.7.b, the devices measure the amount of stored product, the pressure, and the temperature. In this application, tanks can be filled with diverse elements other than fresh oil, such as used oil, wasted liquids, or chemical substances.

The information provided by the devices benefits consumers, suppliers and utility distributors. The monitoring enables consumers to make smarter decisions about how to control this resource, since the supply usage is monitored throughout the day. In addition, it allows the suppliers to better understand the needs of the customers, and the distributors have better tools to manage and monitor their networks.



(a) Smart Greenhouse.
Sensors: Luminosity and humidity.

(b) Tank Monitoring.
Sensors: Temperature and pressure.

Figure 2.7: Energy Harvesting scenarios.

In both scenarios, the amount of renewable energy offered to the nodes is abundant. In addition, using energy harvesting the batteries last longer, reducing the maintenance frequency.

Based on the challenges together with these applications description it is possible to identify a set of desirable functionalities to be executed on the M2M communication. These aspects are presented as follows.

1. **Neutral operation:** Neutral Operation or Energy Self-Sufficient are terms widely used to define energy harvesting efficiency. It means that the nodes must consume the exact amount of harvested energy [Ozel et al., 2015]. To maintain the nodes in neutral operation is a requirement because it guarantees there is no overuse or underuse of the energy on the battery. Doing so, the network lifetime is prolonged and the energy constraints of the nodes can be relaxed during the some periods.

2. **Overhead minimization:** The reduction of control data can impact positively several performance metrics, such as throughput, loss rate, and energy. So, it is important to use less control information inside the message header.

3. **Standard protocol stack:** It is important to promote the development of standard protocols for low-power networks defined by the main standard

bodies, such as IETF. The improvements made on standard protocols have the potential of leverage the adoption of this technology by the industry.

The next section summarizes the content presented in this chapter.

2.5 Summary

This chapter presented the basic concepts and definitions regarding machine-to-machine environments. It began showing the M2M systems are composed of heterogeneous devices that share information to accomplish smart remote control.

Section 2.2 detailed the different scenarios and applications, and the types of traffic produced by machine-to-machine communication. Besides, it discussed the Heterogeneous Hierarchical Architecture, which is a well-known M2M network architecture.

Section 2.3 outlined the most important standards for the understanding of this thesis, related to M2M communication. It emphasized the CoAP protocol, showing the observe and group communication options, and the RPL protocol.

Section 2.4 presented a set of specific scenarios and their communication requirements. The scenarios addressed in this section were divided into two classes, traditional battery with limited energy resources and energy harvesting. In the former, the nodes have severe energy constraints, while in the later, the nodes can replenish the energy resources from the ambient.

The next chapter introduces the study of the related work.

Chapter 3

Related Work

> Study hard what interests you the most
> in the most undisciplined, irreverent
> and original manner possible.
>
> *(Richard Feynman)*

Contents

To design solutions able to improve relevant aspects of Machine-to-Machine communication is a complex task that requires a deep knowledge of different concepts, definitions, and specially the proposed solutions in the literature.

3.1 Introduction

Keeping in mind the main communication requirements based on specific scenarios, this chapter presents a comprehensive literature review. The remainder of this chapter presents in Section 3.2 the study and analysis of the related works.

Next, the research directions are identified in Section 3.3. Finally, Section 3.4 presents a summary of the chapter.

3.2 State-of-the-art

The state-of-the-art study present in this section is focused on reviewing the works able to fulfill totally or partially the communication requirements listed in section 2.4.2. The literature review begins in the next section presenting the solutions able to reduce the communication overhead.

3.2.1 Overhead Reduction

Overhead consumes the resources of the nodes transmitting control information. This is a problem because it consumes resources of the nodes, such as energy, bandwidth, and processing. Overhead is caused by entire messages that contain control information or additional bits introduced in the header of data messages.

In order to reduce the overhead, Lo et al. [Lo et al., 2013] propose a tunnel-based solution scheme which merges data units from several nodes destined to the same tunnel exit-point. This idea avoids repeated headers for the same source-destination pair. It is possible to be executed because when a device sends many data to the same destination node, which is located outside the local network, each protocol involved adds the same header information for each data unit. In contrast to the classic solution, the implemented tunneling solution adds General Packet Radio Service Tunneling Protocol User (GTP-U), UDP and Internet Protocol (IP) headers for several data units. Besides, the solution is designed to queue low priority data units from different nodes until the maximum size of tunneling is reached. The evaluation of this proposal shows a significant overhead reduction on 3rd Generation Partnership Project (3GPP), such as Long Term Evolution, and IEEE 802.16p [Group et al., 2004] technologies.

Taleb et al. [Taleb and Kunz, 2012] also aim to reduce the overhead of M2M communication in cellular networks. The authors of this work show that in M2M communication there is a large number of signaling messages exchanged between the devices and the Mobility Management Entity (MME). One of the main signaling messages is the Tracking Area Updates (TAUs), which is used by the nodes to notify the base-station about their geographic location. The proposed solution defines a time window in the base-station to receive a given number of messages, so all the TAUs received are processed at once. Thus, this solution reduces the message overhead involved in the communication between the base-station and the MME.

Kim et al. [Kim et al., 2017] propose an overhead reduction solution in cellular network, where a concentrator node receives messages from other devices and is responsible for relaying the data to the base station. The concentrator node inserts multiple small packets into a larger packet, reducing the number of accesses between the nodes and the base station.

Tsitsipis et al. [Tsitsipis et al., 2011] and [Tsitsipis et al., 2012] apply a scheme to insert multiple data units inside the same message. In addition, this work focuses on low-power and short-range networks and presents the benefits of the proposed scheme in terms of energy consumption.

Stasi et. al [Di Stasi et al., 2014] focus on reducing the overhead to improve the throughput of multiple path mesh networks. The proposed solution finds queued packets at the MAC layer that can be put together in a single MAC frame. The execution of this solution is done hop-by-hop, where each hop disassembles the incoming packets and then passes the data to the appropriate layer which concatenates packets destined to the same next-hop. The performance evaluation of this proposal shows benefits in terms of throughput in a scenario where the nodes do not have CPU or energy constraints.

The next section presents the solutions able to reduce the redundancy on the data communication.

3.2.2 Data Redundancy Elimination

The basic idea of reducing data redundancy is to apply algorithms on the data collected from the network to eliminate the data similarities, and sometimes repeated information. The approaches able to reduce the redundancy can be classified into:

- **Data Compression Schemes:** These solutions apply compression algorithms on each message to eliminate the redundancy. These schemes do not impact the accuracy of the collected data, since all compressed data can be restored without information loss.
- **Data Aggregation Schemes:** These schemes apply mathematical functions on different messages to reduce the amount of communicated data. These solutions reduces the accuracy of the data, since part of the redundant information cannot be found in the destination side.

In this thesis, a special focus is given to data aggregation schemes, since it is capable of reducing the energy consumption.

Data aggregation schemes became popular on constrained wireless networks, because most of the collected data from nodes is redundant and the final destination is interested on the overall measurement of the nodes. For instance, a group of nodes monitoring temperature of a room tends to measure similar values. Applying data aggregation, instead of receiving different messages with similar temperature, the client receives a single message revealing the average room temperature.

Data fusion schemes differentiate from data aggregation, because the former refers to the process of taking a decision based on the content of one or more message. For instance, a data fusion solution can be used to gather temperature and carbon dioxide to decide if a fire is occurring. Thus, data fusion can eventually result in data elimination, but its final objective is to produce a decision. On the other hand, data aggregation schemes do not involve decision taking procedures, since it uses the message content merely as statistical inputs.

As shown by Jesus et al. [Jesus et al., 2015], the most popular data aggregation schemes apply simple mathematical functions such as sum, average, minimum, maximum and count. These functions are popular because they reveal the general behavior of the measurements, can be computed distributed over the nodes, and have the property of taking many input values and producing a single value.

One of the most representative solutions that applies simple mathematical functions to aggregate data is proposed by Madden et al [Madden et al., 2002]. In this proposal, users can send queries to the network and receive the aggregated response. The solution is designed to allow the network to reply SQL-like queries, which can specify the type of data, the aggregation function, and the location.

Lin et al. [Lin and Chen, 2017] propose a solution that applies the aforementioned data aggregation functions on the data while the messages are traveling from the producer nodes to the destination. Each relay node receives data from neighbors, aggregates the received and the produced data, and transmit it to the next node. This cooperative data aggregation execution is also called in-network aggregation and can be performed in multiple hop communication scenarios.

A large number of works applies aggregation functions to reduce redundancy and aggregate data. Many solutions in this field are coupled with energy efficiency aspects in the routing layer. For that reason, the next section presents routing solutions able to improve the energy efficiency of the communication.

The next section reviews the routing works designed to improve energy efficiency of M2M groups.

3.2.3 Routing-Based Solutions for Energy Efficient Group Communication

Schemes that apply data aggregation or overhead reduction have different energy efficiency according to the set of paths selected by the routing layer. So, many works consider data aggregation, but focused on routing strategies to increase the energy performance of the communication. Figure 3.1 shows three main classes of routing solutions that aim to achieve energy efficiency communication applying data aggregation or overhead reduction in fixed energy budget networks. These solutions are divided in heuristics and Mathematical Optimization.

Heuristics is further classified in Tree-based and Cluster-based approaches. These approaches still are divided in single and multiple group communication. In single group communication, the mechanisms assume the existence of homogeneous communication, which means that all network devices have similar communication settings. For instance, the whole network communicates only one data type (e.g., temperature, humidity, light intensity, gas density, or position), all the geographic areas have the same communication periodicity, or all devices execute the same aggregation function.

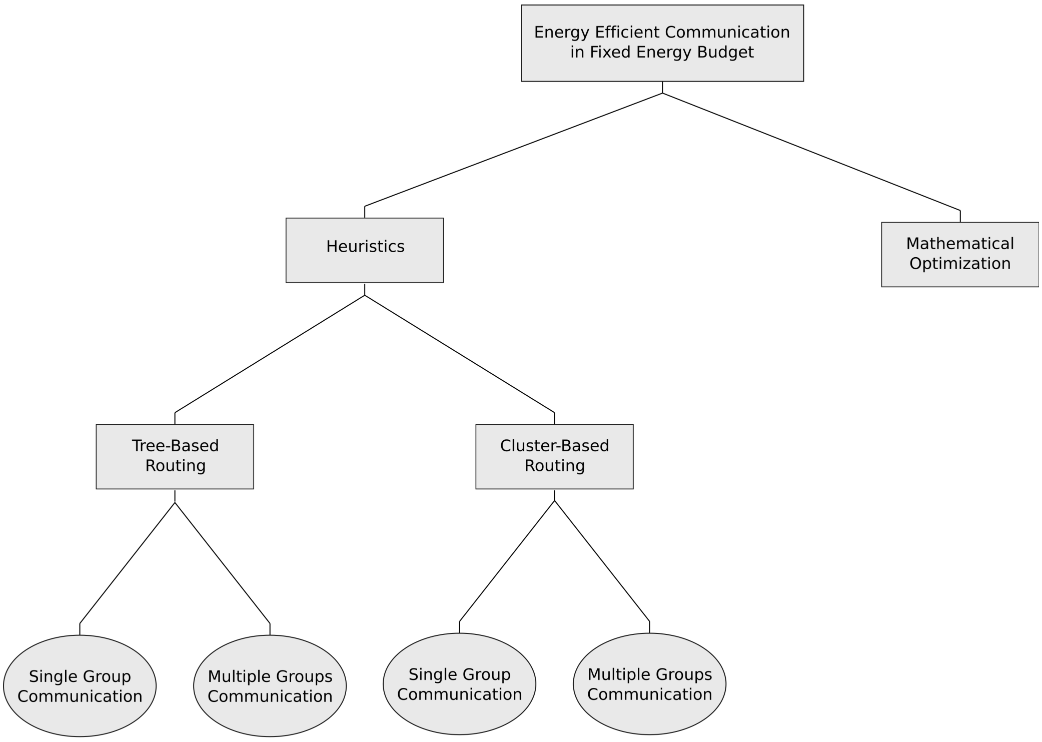


Figure 3.1: Literature classification.

In multiple group communication, the network is divided in set of nodes and each set can have a particular communication setting. The nodes have heterogeneous communication features in terms of data type, aggregation function, or transmission periodicity. The single group communication case becomes a particular case of multiple group communication.

It is important to highlight that all the works described in this section consider fixed amount of energy (i.e. non-replenishable battery). In addition, the term energy efficiency means reducing the energy consumption or maximizing the network lifetime. Network lifetime is defined as the time interval in which all the network nodes have enough energy resource to execute their functions [Azari and Miao, 2017].

The study of the routing solutions starts below, describing the Tree-based solutions for single group communication.

3.2.3.1 Tree-Based Routing for Single Group Communication

The problem of collecting data from energy constrained nodes has been extensively studied. Some works name as covergecast or many-to-one [Long et al., 2017] the type of communication where all network nodes send data to central point. To improve the energy efficiency of this communication type, solutions have been inspired by Graph Theory to build a tree connecting the network and to reduce the costs involved in the data gathering process.

The classic tree algorithms are: Minimum Spanning Tree (MST) and Shortest Path Tree (SPT). Most of the works are based on these algorithms to propose a tree-based solution able to improve the energy efficiency of the communication. For the best understanding of these solutions, it is important to define the terms parent and child. A parent p is the node with h hops to reach the root, while the child is a node connected to p, which has $h + 1$ hops to the root.

The solution proposed by Luo et al. [Luo et al., 2011] computes a Shortest Path Tree (SPT), minimizing the number of children of the nodes with lowest residual energy. The proposed solution only considers the residual energy remaining in the receiver because the energy on the sender is constant during the decision. The link distance between the sender and the receiver is not taken into account, since all the nodes are deployed in coordinates that equalize the distance of the neighbouring nodes. Moreover, this work considers a single group, assumes uniform communication periodicity for all nodes, and all nodes apply aggregation functions of the collected data.

Kacimi et al. [Kacimi et al., 2013] propose a load balancing traffic heuristic to improve the network lifetime. The heuristic balances the amount of traffic on a link, taking into account the transmission power necessary to perform transmissions on that link. The authors show that load balancing strategies are efficient to maintain a fair energy consumption among the nodes. In addition, the work assumes that the nodes are deployed in grid topology and the devices have equal level of energy. This assumption simplifies the problem of maximizing the network lifetime because it assures the parent nodes will be the first nodes to deplete their energy resources, since they will communicate a larger amount of data.

A relevant solution is the Power Efficient Data gathering and Aggregation Protocol-Power Aware (PEDAP-PA) [Tan and Körpeoğlu, 2003]. To achieve maximum network lifetime, PEDAP-PA considers the wireless links as edges of a graph and assigns a weight to each edge. The weight is associated with the energy cost involved in the communication of a pair of nodes. PEDAP-PA also includes in the weight formula the sender residual energy, so the more residual energy a sender has, the less is the weight. To find the best data aggregation performance in terms of network lifetime, PEDAP-PA computes the Minimum Spanning Tree (MST) over a weighted graph. As result, PEDAP-PA shows better performance than Low Energy Adaptive Clustering Hierarchy (LEACH) [Heinzelman et al., 2002] and Power-Efficient GAthering in Sensor Information Systems (PEGASIS) [Lindsey and Raghavendra, 2002], which are well-known clustering solutions.

PEDAP-PA has two main drawbacks, (i) the algorithm to estimate a link cost does not consider the residual energy of the receiver node, and (ii) the weights are computed every 100 communication rounds regardless the amount of consumed energy. To solve these problems, Energy Efficient Spanning tRee (EESR) [Hussain and Islam, 2007] proposes a dynamic scheme, which considers the residual energy of the receiver node, and it recomputes the weights in dynamic periods. Other PEDAP-PA improvement is presented by Lin et al. [Lin et al., 2010], providing an algorithm that iteratively removes edges from nodes that are spending more energy and adds new edges to nodes that are spending less energy. In addition, Tan et al. [Tan et al., 2011] present the Localized Power-Efficient Data Aggregation Protocol (L-PEDAP) solution, which is a distributed version of PEDAP-PA.

Another solution, proposed by Mottola et al. [Mottola and Picco, 2011], called MUSTER, is designed for single group scenarios with multiple sinks. As these scenarios involve multiple sources and destinations, it is necessary to compute a set of trees to enable the communication. MUSTER is a distributed solution, where each node exchanges control information with the parent candidates to compute (i) the routing quality and (ii) the expected lifetime of the parent candidates. The routing quality is calculated for each parent n considering the reliability of n, the number of paths passing through it, and the number of sinks it is sending data. The main objective of the solution is to reduce the number of nodes involved in the communication and maximize the path overlapping to take advantage of the data aggregation.

Next, the tree-based solutions designed for multiple groups are described.

3.2.3.2 Tree-Based Routing for Multiple Group Communication

Regarding multiple group communication, where each group has a different communication setting, the solution proposed by Xu et al. [Xu et al., 2009] builds a Shortest Path Tree connecting the nodes and all parent nodes execute aggregation functions. It starts building the tree with the root, and in each iteration, it adds the shortest path to the nearest data producer device. The process continues until all source nodes have been included in the tree. The focus of this work is to study the latency of the proposed tree, trying to find a communication scheduler to minimize the latency. Thus, this work does not study how the proposed tree impacts the network lifetime.

The solution proposed by Li et al. [Li et al., 2006] is designed to aggregate data from groups that monitor different targets and communicate in accordance with a particular communication periodicity. This work considers that over time the groups can become active to produce and communicate data. Otherwise, the group remains inactive. The solution computes a Minimum Spanning Tree which includes all devices belonging to active groups, and reduces the number of links belonging to inactive groups. In this solution, the internal group traffic is forwarded to the sink node via the shortest path without applying aggregation functions or any other scheme to reduce the overhead.

Lee et al. [Lee et al., 2010] propose a framework to enable the network to perform group-by aggregation queries. The groups are formed according to the geographic regions. The solution selects a leader node for each group. The traffic sent from the members to the group leader is routed by a tree. The member nodes applies a first round of Haar wavelets on the produced data and sends it to the leader. The leader node receives wavelet coefficients from member nodes and computes the second round of Haar wavelets. After the group leader compresses the data, it is sent to a central point.

The next section presents the solutions that use clustering algorithms for single group communication.

3.2.3.3 Cluster-Based Routing for Single Group Communication

Cluster-based routing has been largely used to improve energy efficiency of low-power networks. Most the cluster solutions have a scheme to select cluster heads, which aggregates the data received from the cluster members.

The major part of the cluster-based routing solutions that apply data aggregation or other overhead reduction scheme considers the network nodes as members of a single group. Under this idea, the cluster solution can divide the network using its own metric without considering any existing group defined by the user or the applications. Besides, in cluster-based solutions, the cluster head is the main point to perform aggregation or overhead reduction, creating a 3-depth topology, namely cluster members, cluster head and base-station.

Low-Energy Adaptive Clustering Hierarchy (LEACH) [Handy et al., 2002] is one of the most popular cluster-based proposals. LEACH selects a new cluster head periodically and assumes that every node is capable of being a cluster head. When a cluster head is elected, it broadcasts a message to all the other nodes. All the cluster members that receive an advertisement must select the sender as cluster head. When a node receives multiple advertisement, the receiver must select one cluster head using the signal strength or residual energy. In addition, LEACH assumes that the cluster heads are able to transmit directly to the sink node.

The Hybrid Energy-Efficient Distributed clustering approach (HEED) [Younis and Fahmy, 2004] selects the cluster heads based on the residual energy combined with a parameter that combines the node proximity with its neighbors. This solution also considers the size of the cluster as a parameter to define the number of cluster heads in the network and ensures that all nodes have wireless connectivity.

The next section continues the literature review, showing the cluster-based solutions that consider multiple groups.

3.2.3.4 Cluster-Based Routing for Multiple Group Communication

The solution proposed by Al et al. [Al-Karaki et al., 2009] provides a hierarchical cluster-based approach, called Grid-based Routing and Aggregator Selection Protocol (GRASS), to find the paths and the points of data aggregations. This approach considers a scenario where the network nodes are tracking different targets and for each target there is a particular group of monitoring nodes. Each group has Local Aggregators (LA) and only the cluster head performs aggregation functions. To perform the aggregation of LA data, a second level of aggregation is done by the Master Aggregators (MA). The problem is reduced to finding a set of LA and MA such that the network lifetime is maximized. The authors provide optimal and near-optimal solutions for this problem. However, GRASS assumes that all nodes, even the nodes tracking different targets (different groups), have an uniform group setting (e.g. same period of communication and same aggregation function). Although GRASS considers the existence of multiple groups, data is aggregated regardless the group where it was produced.

Considering heterogeneous groups, Choi et al. [Choi and Chung, 2013] propose a solution to aggregate data from multiple geographic targets. The proposed solution, called REQUEST$^+$, allows the monitoring of multiple geographic regions defined previously by the user. REQUEST$^+$ is designed to aggregate data even when the targets overlap, which means that a node is able to monitor more than one target. REQUEST$^+$ builds a cluster-based topology to perform the aggregation. All the nodes inside a particular target area belong to the same cluster, and each cluster has a cluster head. This solution computes a tree, where the root is the cluster head, to define the paths between the regular nodes and cluster head. This tree is computed to minimize the number of hops for the intra-cluster traffic.

After presenting the cluster-based solutions, the next section describes the mathematical approaches that apply data aggregation or overhead reduction for energy efficient communication.

3.2.3.5 Mathematical Optimization Approaches

The mathematical models that determine the optimal flow of data have the benefit of achieving the maximum or minimum objective function, which can be defined in terms of network lifetime or energy consumption. The optimal solution becomes a reference or an upper-bound, since the model simplifies many aspects to make the problem solvable.

One of the disadvantages of optimal solutions is that, in most cases, it is centralized and requires the full knowledge of the network. In addition, the computation effort required to find the optimal solution is high, mainly when the problem involves large scale. Some techniques enable the decomposition of the centralized models into distributed, but they increase largely the complexity of the solution.

Among the works that address the problem of maximizing the network lifetime using data aggregation, Kalpakis et al. [Kalpakis et al., 2003] formulate a Linear Programming (LP) model to carry this out in single group communication scenarios. This LP model is one of the most important references for maximizing the network lifetime by means of data aggregation. However, it does not consider either multiple groups or schemes to promote overhead reduction. In the model proposed by Kalpakis et al. [Kalpakis et al., 2003], the aggregation can be carried out without knowing the data producer, which means the LP formulation is based on the single-commodity network flow problem. In the case of multiple groups, the aggregation is performed by considering each group as a different data producer, and thus the problem of determining the maximum network lifetime should be addressed on the basis of the multi-commodity network flow problem.

Hua et al. [Hua and Yum, 2008] generalize the formulation of Kalpakis et al. [Kalpakis et al., 2003], addressing the problem of maximizing the network lifetime and considering other aggregation operations rather than simple mathematical functions. However, the authors were only concerned with finding efficient heuristics for single group applications.

Bicakci and Tavli [Bicakci and Tavli, 2010] investigate communication strategies for prolonging the network lifetime in multi-domain wireless sensor networks also through linear programming. The authors consider a scenario of cooperative multi-domain networks deployed in the same physical location, where a different authority manages each domain. The idea of a cooperative multi-domain network is similar to the concept of multi-group communication. However, this work does not apply any data aggregation approach on the network traffic.

So far, the proposed solutions are designed for nodes with fixed energy budget, which means the nodes have a fixed, pre-determined amount of energy to spend. In the next section, the solutions consider that the nodes have means to renew their reserves.

3.2.4 Neutral Operation in Renewable Energy Budget

The term Neutral Operation or Energy Self-Sufficient is widely used to define energy efficiency in harvesting scenarios. These terms mean that a node only consumes the amount of energy that has been harvested. When a node is in neutral operation, there is no overuse or underuse of the energy on the battery. Therefore, the main objective of these solutions is to achieve neutral operation. To achieve this objective, the works define manners to regulate the amount of energy that can be consumed. The next section describes the solutions based on the Single Cycle of Charge and Duty.

3.2.4.1 Single Cycle of Charge and Duty

Some solutions rely on the Single Cycle of Charge and Duty (SCCD) to achieve the neutral operation of the network. SCCD allows a node to assume only two states over the time, namely charging or active state. In the charging state, the node collects energy from the ambient, while in the active state, the node is allowed to execute communication actions, such as data transmission and reception. According to SCCD, every node remains in the charging state until it stores the amount of energy to perform a single data reception and transmission.

Relying on SCCD, Eu et al. [Eu and Tan, 2012] propose an opportunistic routing solution, in which each transmitted message can be received by multiple neighbors. However, for the case of multiple receptions, only the closest node to the sink forwards the message. As this solution is based on opportunistic routing, it has the advantage of not requiring synchronized communication, and increases the reliability of the communication. On the other hand, it requires schemes to avoid message duplication, and it is designed for one-to-one communication, which makes this solution not suitable for data aggregation.

Also based on SCCD, Suh et al. [Suh et al., 2009] propose a solution that addresses routing and sleep scheduling. This solution selects a set of relay nodes, which are the nodes that receive data from the neighbors and send it directly to the sink. According to this proposal, the maximum number of communication hops is 2. Besides that, the sleep/wake up timings are computed in a centralized fashion. Although this solution can be used for many-to-one communication, it has the following drawbacks: (i) the constrained number of communication hops undermines the execution of data aggregation in medium and large scale scenarios, (ii) this solution assumes that during the wake up time there is no energy to be harvested, and (iii) only a very small topology is evaluated, where the nodes are distributed in line. Another drawback of this solution is that it does not consider any standard protocol in the communication stack.

Some works do not restrict the activities like SCCD. Instead, there are solutions that find manners to regulate the amount of traffic to control the energy consumption. These solutions are presented below.

3.2.4.2 Regulating the Amount of Traffic

A solution on neutral operation must consume more energy when there are more energy resources on the reserves. It is well-known that the amount of consumed energy is influenced by the amount of traffic communicated by the nodes. An effective idea is to regulate the amount of traffic traveling on the network in accordance to the amount of energy available.

Following this idea, Peng et al. [Peng and Low, 2013] propose a query-based routing protocol inspired on Directed Diffusion [Intanagonwiwat et al., 2003]. According to this solution, the sink node floods the network with interest messages and all regular nodes send a reply message to the sink. Then, the sink node sends another message, called Reinforcement Interest, which specifies the communication data rate for the regular nodes. However, the regular nodes only accept this data rate if it does not compromise the amount of energy available in their energy reserve. The solution considers the amount of energy harvested during a whole day as parameter to estimate the amount of available energy. Thus, this solution introduces a sort of admission control for controlling the energy consumption in energy harvesting.

Lattanzi et al. [Lattanzi et al., 2007] propose a solution that finds a traffic workload (i.e. communication rate) able to maintain the energy consumption sustainable. It is developed for synchronous and constant-periodic monitoring applications, where all the nodes provide data at the same communication rate. However, this solution assumes that the batteries can store unlimited amount of energy, preventing the nodes to experience energy overflow. Although this work was designed for many-to-one communication, the algorithm that computes the traffic workload does not take into account the influence of data aggregation or overhead reduction scheme on the amount of traffic.

Gao et al. [Gao et al., 2014] address data compression in energy harvesting networks. This work proposes a reactive solution that dynamically changes the compression level of the nodes according to the harvested energy. The authors propose an in-network approach that assigns low compression level for the nodes that have high residual energy and low communication costs. Besides, each node starts with the minimum compression level, and increases it gradually. The solution stops to increase the compression level if the node's energy consumption exceeds the energy extracted from the ambient. This solution has the following drawbacks: (i) it is a centralized solution, and (ii) as the data compression level depends solely on the amount of harvested energy, the authors do not show how the proposed solution computes the compression level when there is no harvested energy from the ambient (e.g. during periods of no or low solar irradiation).

Assuming unreliable wireless links, Zhang et al. [Zhang et al., 2011a] propose a solution that addresses aspects of link scheduling and data aggregation in energy harvesting networks. In this work, the network traffic is aggregated by means of a tree, which is computed based on the Weighted Connected Dominating Set (WCDS). The proposed solution finds the tree assigning a weight for each node and computes the Maximum WCDS, where the weight is a function of the harvested energy. In this approach, a parent node receives and aggregates all messages into a single message and transmits to its parent. However, this solution does not vary the aggregation level, since a node always aggregates all messages into a single message, achieving the maximum aggregation level.

Jeong et al [Jeong et al., 2016] also propose a solution for regulating traffic load in energy harvesting networks. In this solution, before the periodic communication is executed, a node estimates its remaining energy. In case the estimated remaining energy is expected to cause overflow on the energy reserve, the node transmits data to avoid wasting of energy. If the energy reserve is expected to be depleted, the node does not transmit any data. Although this work considers data aggregation, it does not use different data aggregation levels to communicate different traffic loads. In addition, Jeong et al [Jeong et al., 2016] rely on a linear battery model to estimate the battery level, which is a very unrealistic assumption.

Next, some other relevant solutions for neutral operation are described.

3.2.4.3 Other Relevant Approaches

Xiao et al. [Xiao et al., 2013] present a solution that adapts the Low Energy Adaptive Clustering Hierarchy (LEACH) protocol for energy harvesting scenarios. The proposed solution changes the traditional LEACH algorithm responsible for selecting the cluster heads. This proposal considers the predicted energy harvested and the current residual energy for selecting cluster heads. However, this solution does not address the data aggregation aspects.

Zeng et al. [Zeng et al., 2009] propose two routing metrics designed for energy harvesting networks with unreliable wireless links. These metrics are used to select the best forwarder neighbor, and they take into account the neighbor's location, residual energy, short-term energy harvesting rate, energy consuming rate, and wireless link quality (i.e. frame delivery ratio). Although this solution is evaluated with many-to-one traffic, it does not consider aspects related to data aggregation.

After the analysis of the literature, the next section discusses the relevant open issues.

3.3 Open Issues

This section aims to present the open issues found after studying the requirements and the state-of-the-art. The identified open issues in this section are the research directions used as guide to design and develop the solutions contained in this thesis. Firstly, the open issues related to multiple and heterogeneous group communication are presented in Section 3.3.1. After that, Section 3.3.2 describes the open issues regarding neutral operation in energy harvesting scenarios.

3.3.1 Multiple and heterogeneous Group Communication

Tables 3.1 and 3.2 show how analyzed works fulfill the communication requirements listed in Section 2.4.2. Table 3.1 presents the works focused on single groups, while Table 3.2 shows the solutions related to multiple groups. As can be noticed, three works propose schemes to reduce the overhead, while most solutions only apply data aggregation to reduce the redundancy. Heterogeneous group communication is not widely explored by the approaches.

Table 3.1: Literature related to Single Group Communication.

Work	**Redundancy Reduction**	**Overhead Reduction**	**Energy Efficiency**
[Handy et al., 2002]	✓	✗	✓
[Tan and Körpeoğlu, 2003]	✓	✗	✓
[Kalpakis et al., 2003]	✓	✗	✓
[Younis and Fahmy, 2004]	✓	✗	✓
[Hussain and Islam, 2007]	✓	✗	✓
[Hua and Yum, 2008]	✓	✗	✓
[Lin et al., 2010]	✓	✗	✓
[Tan et al., 2011]	✓	✗	✓
[Mottola and Picco, 2011]	✓	✗	✓
[Luo et al., 2011]	✓	✗	✓
[Tsitsipis et al., 2012]	✗	✓	✓
[Kacimi et al., 2013]	✓	✗	✓
[Lo et al., 2013]	✗	✓	✓
[Kim et al., 2017]	✗	✓	✓

Table 3.2: Literature related to Multiple Group Communication.

Work	**Redundancy Reduction**	**Overhead Reduction**	**Energy Efficiency**
[Li et al., 2006]	✓	✗	✓
[Xu et al., 2009]	✓	✗	✗
[Al-Karaki et al., 2009] *	✓	✗	✓
[Lee et al., 2010]	✗	✗	✓
[Bicakci and Tavli, 2010]	✗	✗	✓
[Choi and Chung, 2013]	✓	✗	✓
[Di Stasi et al., 2014]	✓	✗	✗

* Homogeneous groups.

The study of the state-of-the-art shows that multiple and heterogeneous group communication in M2M scenarios is an open research field. Keeping in mind the requirements of this type of scenario, two specific open issues are presented as follows:

- To improve the energy efficiency of heterogeneous group communications using mechanisms to reduce the data redundancy and the overhead of the communication.

- To improve the manageability of the groups is also an open aspect, since the tasks related to creation, modification, and deletion of a large number of nodes and groups are not addressed by the solutions.

The next section presents the open research aspects in energy harvesting networks.

3.3.2 Neutral Operation in Energy Harvesting Scenarios

Table 3.3 shows the main features of the solutions proposed for energy harvesting scenarios. Most of the solutions aim to achieve neutral operation, which means to minimize the battery depletion and the waste of energy. To achieve this objective, part of these solutions consider mechanisms to regulate the amount of traffic on the network, since more traffic results in higher energy consumption, and vice-versa. However, none of the works propose solutions considering the standard protocol stack for low-power networks, mainly the routing and application protocols, RPL and CoAP, respectively. Instead of considering the standard protocols, most of the approaches rely on theoretical models to formulate the behavior of some layers. Besides, only a reduced number of works consider schemes to reduce the traffic overhead.

Table 3.3: Literature related to Energy Harvesting Scenarios.

Work	**Traffic Regulation**	**Overhead Reduction**	**Neutral Operation**
[Lattanzi et al., 2007]	✓	✗	✗
[Zeng et al., 2009]	✗	✗	✓
[Suh et al., 2009]	✓	✗	✓
[Eu and Tan, 2012]	✓	✗	✓
[Zhang et al., 2011a]	✓	✗	✓
[Xiao et al., 2013]	✗	✗	✓
[Peng and Low, 2013]	✓	✗	✓
[Gao et al., 2014]	✓	✗	✓
[Jeong et al., 2016]	✓	✗	✓
[Kim et al., 2017]	✗	✓	✓

Given the analysis of the state-of-the-art related to energy harvesting scenario, the following open issues can be listed:

- To achieve neutral operation in low-power networks applying mechanisms to regulate the amount of traffic traveling over the network.
- To promote extensions in the standard protocols stack in order to improve their efficiency in energy renewable scenarios.
- To develop solutions to reduce the traffic overhead, since these scenarios can suffer from traffic congestion.

To finalize this chapter, the next section presents a summary.

3.4 Summary

This chapter presented a comprehensive discussion with the final objective to identify the gaps in the current state-of-the-art. It presented a set of challenges in Section 2.4. Besides, in this section, the M2M scenarios were narrowed to allow the identification of communication requirements.

After the discussion around the challenges and identification of the requirements, Section 3.2 showed the state-of-the-art study, with the aim to find how the existing literature solutions solve or fulfill the challenges and the selected communication requirements.

To accomplish the objective of this chapter, Section 3.3 presented an analysis about the open issues and research directions. This analysis is critical for the design of the proposed solutions detailed in the next chapters of this thesis.

The next chapter presents the proposed mechanisms for M2M Group Communication.

Chapter 4

Mechanisms for Energy Efficiency in M2M Group Communication

> The saddest aspect of life right now is that science gathers knowledge faster than society gathers wisdom.
>
> *(Isaac Asimov)*

Contents

Heterogeneous and multiple group communication is an important niche in the general machine-to-machine landscape. In this field, groups are formed by devices that have few kilobytes of memory (Random

Access Memory (RAM) and Read-Only Memory (ROM)), slow processing, and reduced bandwidth. In addition, most of the devices are battery-powered and have limited lifetime. Given these constraints, data aggregation and overhead reduction schemes show large potential to improve the group communication. However, the use of these schemes unveils the need for adaptive mechanisms to leverage the energy efficiency. Thus, this chapter aims to present the design and evaluation of mechanisms able to support data aggregation and overhead reduction in order to improve group communication.

4.1 Introduction

Some of the requirements for group communication scenarios are: (i) Eliminate data redundancy, (ii) Minimize overhead, and (iii) Support heterogeneous groups. The fulfillment of these requirements would drive the machine-to-machine communication to the efficient use of the resources and also to deal with the massive number of nodes not individually, but as groups.

Although some stadardization bodies have made advances in this direction, the literature review shows that few works fulfill partially all these requirements. So, in order to fill this gap, this chapter presents:

- General scheme that jointly applies data aggregation procedures to eliminate data redundancy and reduce overhead in group communications.
- Two tree-based heuristics to execute the proposed data aggregation procedures over the nodes to prolong the network lifetime.
- The formulation of the network lifetime upper bound and the performance comparison between the heuristics and the optimal solution.
- Testbed experiments involving the proposed data aggregation procedures.

To present all these contributions, this chapter is structured as follows. Section 4.2 presents the proposal of a general scheme that performs data aggregation and overhead reduction. Section 4.3 presents the following heuristics: Data Aggregation for Multiple Groups (DAMiG) and Two Tier Aggregation for Multi-target Applications (TTAMA). Section 4.4 models the network lifetime upper bound and compares it to the heuristic solutions. Finally, Section 4.5 presents the testbed experiments.

4.2 Data Aggregation and Overhead Reduction Scheme

In the literature, there are different generic aggregation functions and several manners to combine the functions to achieve data redundancy elimination and overhead reduction. A manner to achieve both objectives in group communication is to design a scheme with multiple tiers, where each tier is restricted to aggregate data from a specific set of nodes and to use a particular type of aggregation functions. For instance, while one aggregation tier aims to reduce the level of data redundancy, a second tier aims to reduce the communication overhead.

The general proposed scheme, named Two Tier Aggregation (TTA), uses the following two techniques on the network traffic:

- **Payload Aggregation for Redundancy Elimination:** applies statistical functions to aggregate the payload information into a single value. As depicted in Figure 4.1, using the *average* function, received payloads containing the values *20* and *10* can be aggregated into a single payload containing *15*. The aggregation functions eligible to be used as payload aggregation function are average, maximum, minimum, count, and sum, since they have the property to produce one output value regardless the number of input values.

Received Message 1:	Header	20
Received Message 2:	Header	10
Aggregated Message	Header	15

Figure 4.1: Payload Aggregation.

- **Payload Concatenation for Reduced Overhead:** produces a message that has a single header and multiple attached payloads, which means that the size of the messages is variable. As shown in Figure 4.2, received payloads are conserved, but it avoids sending one header for each payload.

Received Message 1:	Header	20	
Received Message 2:	Header	10	
Aggregated Message	Header	10	20

Figure 4.2: Payload Concatenation.

To understand how these two techniques aggregate the traffic produced by multiple groups, it is necessary to make a distinction between Internal and External Group Traffic. When a particular group is taken as a reference point, *Internal Group Traffic* is the set of messages originating from its members, while *External Group Traffic* is the set of messages produced by other groups.

Seeking to fulfill the requirements of multiple group communication, the main idea of TTA is to apply Payload Aggregation on Internal Group Traffic and Payload Concatenation on External Group Traffic.

Thus, Two Tier Aggregation uses two rules: (i) When the data is in transit inside its producer-group, it is stored in the message as the primary payload and will be aggregated by means of simple mathematical calculations (i.e. the Payload Aggregation), such as average, maximum, minimum, count, and sum; (ii) If the data goes outside the producer group, it is maintained intact as a secondary payload, which is attached to the primary payload (i.e. the Payload Concatenation). Thus, the messages produced by the proposed data aggregation approach have a primary payload and might have multiple secondary payloads attached.

By using Payload Aggregation, TTA can avoid the problem of similar or even repeated messages being sent by the group members. In addition, by means of Payload Concatenation, TTA eliminates redundant headers from the External Group Traffic, and produces messages with a single header and multiple payloads. This saves energy resources that would otherwise be spent on message headers, without impairing the accuracy of the communication, since a node can suppress the headers of messages produced by another group, and thus keep the payload content intact.

Figure 4.3 illustrates Two-Tier Aggregation. This figure shows a particular communication event when a node, which belongs to group n, aggregates Internal and External Group Traffic, by applying the payload aggregation to the Internal Group Traffic and concatenating the payloads derived from the external groups. As can be observed, the produced aggregated message has a single *intra* payload resulting from the payload aggregation operations (e.g. average, maximum, minimum, count, and sum), but every *external* payload is preserved.

Payload aggregation is only performed on internal group traffic because each group must have a particular communication setting, such as a data aggregation function, communication periodicity, and data types. For instance, group A can be defined by the client to communicate the maximum temperature of a room, while group B gives information about the average luminosity in an office. Thus, it is not possible to compute payload aggregation of groups A and B due to the different settings (i.e. aggregation functions) and the data types. Another reason to perform payload aggregation only on internal group traffic is that the nodes of a particular group are usually located in the same geographic area, tending to present high redundancies on their data. On the contrary, groups located in different areas are not likely to present such high data redundancy.

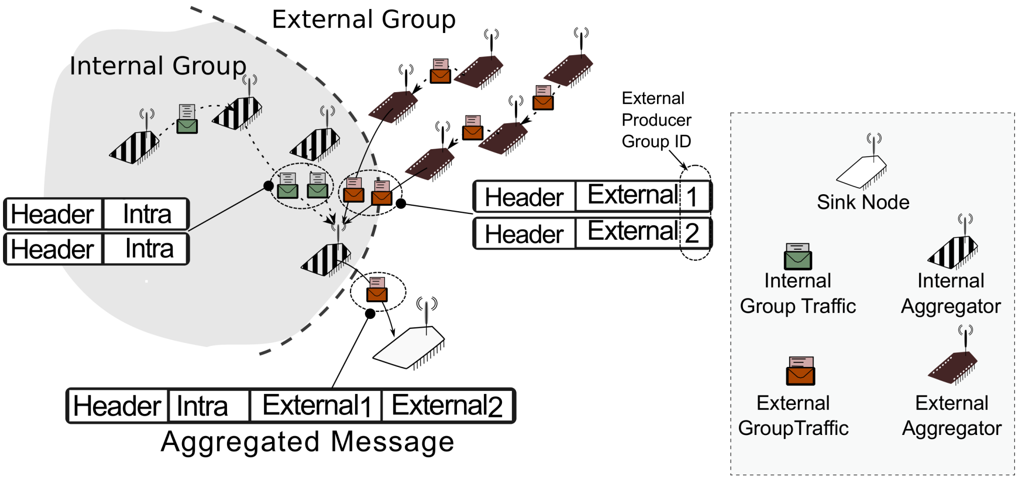


Figure 4.3: Two Tier Aggregation.

To illustrate payload concatenation, let us assume the perspective of a node i. Note that, for node i, the payload concatenation consists on the following two steps: (i) to drop the headers of all messages received from neighbor j, and store only the payloads; (ii) to generate a new message containing a single header and multiple payloads. The number of payloads on a single message corresponds to the number of payloads received from external groups plus one payload produced by the group. As the payloads have a fixed size, it is possible to define rules to identify the internal group payload (e.g. the payload belonging to the internal group is always the one placed after the end of the header). Thus, to perform payload aggregation on the internal group traffic, a receiver i verifies the address of the sender j. If the receiver (node i) and the sender (node j) belong to the same group, then the receiver uses a payload aggregation function, such as average, to aggregate its own data with the internal group payload. Otherwise, it adds its own data as the first payload of the new message and transmits.

The next section presents the proposed heuristics designed to use the Two-Tier Aggregation scheme.

4.3 Heuristics for Efficient Group Communication

The Two-Tier Aggregation is able to eliminate the data redundancy and reduce the overhead present in the traffic produced by groups. However, this scheme needs to be further specified in order to be executed distributed over the network. As the state-of-the-art shows, tree-based approaches are widely used to perform in-network data aggregation schemes. As the computed trees provide support for the aggregation execution, the terms *aggregation tree* and *tree* are used interchangeably.

This section proposes two tree-based heuristics able to aggregate data from multiple groups and keep the overhead at low levels. The first approach is named Data Aggregation for Multiple Groups (DAMiG) and the second is called Two-Tier Aggregation for Multi-target Applications (TTAMA). Both approaches execute Two-Tier Aggregation in group communication scenarios, but TTAMA achieves better network lifetime, since it proposes an improved algorithm to compute the aggregation tree.

The next section presents how the system is modeled, in special the terminology and energy consumption model.

4.3.1 System Model

This section presents the theoretical model used to represent the network and also introduces the equations used to compute the consumed energy and network lifetime.

4.3.1.1 Terminology and Network Model

In order to define a common set of basic terms with unambiguous meanings, Table 4.1 presents key terms used in this chapter.

Table 4.1: Terminology Definition

Term	**Definition**
Group	A set of connected nodes in the network.
Group Settings	It defines the aggregation function, the objective metric, the communication periodicity, and the tolerable delay of a group.
Concurrent Groups	Two groups are concurrent when both groups communicate in the same communication round.
Communication Round	A communication round occurs when all nodes in the network deliver a single own produced payload to the sink.
Node Lifetime	It is the total number of communication rounds that can be executed before the first occurrence of battery depletion.

The notations used in this section are listed in Table 4.2. The network is represented by a graph $G = \{V, A\}$, where $V = \{1, 2, 3...n\}$ is the set of nodes and A is the set of edges. We assume that G is a connected graph, and the edges represent wireless connections. Every node $i \in V$ has the same wireless range, which is denoted by R. There is an edge between two nodes $i, j \in V$ if and only if the Euclidean distance is $d_{i,j} \leq R$.

The network has a single sink node, denoted by $s \in V$, which is the final

Table 4.2: Notation.

Symbol	**Definition**
$G(V, A)$	The graph of the network, V represents the set of nodes, and A corresponds to the set of edges.
R	Wireless range.
$d_{i,j}$	Euclidian distance between two $(i, j \in N)$ neighbour nodes.
T	Set of targets to be monitored.
A_k	Set of devices that monitor the k-th target.
D_l	Data readings produced by the nodes during the l-th communication round.
s^k_{agg}	Aggregation Function of A_k.
s^k_{obj}	Objective of A_k.
s^k_{freq}	Communication Periodicity of A_k.
s^k_{delay}	Delay Tolerance of A_k.
e_{elec}	Energy necessary to run a single bit.
e_{amp}	Energy necessary to amplify a single bit.
Q_i	Set of children nodes of node i.
w^k_i	Data reading after perform data aggregation function.
$E_{res}(i)$	Residual energy of node i.
$E_{round}(i)$	The energy consumed by node i during a single round.
L_i	The lifetime estimation of node i.
NL	The network lifetime time estimation.

destination of all messages. The set of regular nodes (i.e. non-sink nodes) is denoted as $N = V - \{s\}$. Every node $i \in N, i \geq 1$ is powered by a non-replenishable battery. The amount of residual energy in the battery of node i is denoted by $E_{res}(i)$. We assume that all the nodes are able to determine their own residual energy, and execute Payload Aggregation and Concatenation. Unlike the ordinary nodes, the network gateway has a constant energy supply. In addition, we assume that the nodes have static geographic positions, as it is the case of smart metering and smart grid applications.

The communication round is the unit of time used, and it is defined as the time interval sufficient to deliver to the gateway the data readings sent by all nodes interested to communicate in that round. For instance, it can be supposed that in round 3 only group A_2 has communication settings demanding the communication of data readings. In round 4, A_5 and A_6 are the groups interested to communicate data. According to the definition of a communication round, round 4 only begins when the data readings from group A_2 are delivered to the network gateway, and ends when all the data from groups A_5 and A_6 have been received by the gateway.

During a particular communication round, if a device communicates its own data, it is called *source node*, otherwise it is named *non-source node*. It should be noted that the non-source nodes may participate in the multi-hop communication, since in some network topologies the source nodes might not have direct wireless connection with the network gateway.

In the network there are multiple targets $T = \{t_1, t_2, t_3, ..., t_m\}$ to be monitored, and $\forall t_k \in T, \exists\ A_k \subset N$, where $k \in \{1, 2, ..., m\}$ and $|A_k| > 0$. Therefore, A_k represents a set of nodes that monitor the target t_k. We assume that a node might belong to only one group A_k, which means that there is no overlap among the nodes that are monitoring different targets.

The next section continues the general definitions, presenting the energy consumption model.

4.3.1.2 Energy Consumption Model

The nodes have three energy states, transmission (Tx), reception (Rx), and sleep. The Equations 4.1 and 4.2 denote the energy necessary for node i to transmit b bits to the neighbour, and the energy spent by the node $i \in N$ to receive the same b bits from j.

$$Tx_i = e_{amp} d_{i,j}^2 b \tag{4.1}$$

$$Rx_j = e_{elec} b \tag{4.2}$$

Besides, it is well-known that the transition between the energy states consumes power. Thus, s_{trans} denotes a single transition from Rx or Tx to sleep, or vice-versa, and according to Ma et. al [Ma et al., 2009] we consider $s_{trans} = 21\mu J$. Owing to the high energy costs incurred in these transitions, the solutions implemented and evaluated use TDMA-based like the MAC layer used by Ma et. al [Ma et al., 2009], which is a collision-free protocol that minimizes the transitions from one energy state to another. When this protocol is used, the nodes involved in the communication will spend $2 * s_{trans}$ in a single communication round. This is because the nodes change once from sleep mode to Tx or Rx mode, and afterwards, change once again from Tx to sleep mode. In addition, since it can be assumed that the devices are able to maintain perfect clock synchronization, the energy spent on overhearing and idle states is not considered.

Let $d_{i,j}$ denote the Euclidian distance of two neighbour nodes as $i, j \in N$, and the communication rounds as l. In addition, it is possible to define the data readings produced by the nodes in a particular l as $D_l = \{w_i^k | k \in \{1, 2, ..., m\}, i \in N\}$. Besides, let the set of children of node i be denoted as $Q_i = \{1, 2, ..., z\}$. Then, w_i^k represents the data to be transmitted by the node i after aggregating its own data with $w_q^k, q \in Q_i$. Thus, in a communication round, the energy consumption of a node $i \in N$ is given by the following function:

$$E_{round}(i) = e_{amp} * d_{i,j}^2 * w_i^k + e_{elec} * w_q^k + 2 * s_{trans} \tag{4.3}$$

The first two terms of Equation 4.3 represent the energy consumption related to the data reception and data transmission, and the third term measures the energy consumption caused by the transitions between different energy states. As it can be seen, the energy costs related to the processing and sensing activity are not taken into account in this model.

The next section describes the problem to be addressed by the heuristics solutions.

4.3.1.3 Problem Statement

Based on the energy consumption of node i during a communication round (given by Equation 4.3), the residual energy $E_{res}(i)$, and the communication periodicity of its group, it is possible to estimate the remaining lifetime of i. The following function gives this estimate:

$$L_i = E_{round}(i)/E_{res}(i), i \in N, k \in \{1, 2, ..., m\} \tag{4.4}$$

Equation 4.4 reveals how many communication rounds the battery of a node will be able to provide energy, which means the remaining node lifetime. The network lifetime is defined as the communication rounds between the first data communication and the time when the first node depletes its energy [Karakus et al., 2013]. Hence, by knowing the node lifetime, we can formulate the following function to compute the network lifetime.

$$NL = min(L_i), \forall i \in N \tag{4.5}$$

Thus, the problem of maximizing the network lifetime can be defined as the following minimax function:

$$min\ max[1/(L_i)], \forall i \in N \tag{4.6}$$

The energy consumption model (Equation 4.3) and consequently the network lifetime formulation (Equation 4.5) are based on the amount of data transmitted (w_i^k, $i \in N, k \in \{1, 2, ..., m\}$) and received ($w_q^k$, $q \in Q_i$). The data aggregation model is a key element in saving energy and increasing the network lifetime, since it is responsible for determining w_i^k and w_q^k.

The problem to be addressed by the heuristics is to find a set of trees able to extend the network lifetime using Two-Tier Aggregation on traffic, approximating the maximum network lifetime.

After showing the notation, the energy consumption model and the problem statement, the next section introduces a tree heuristic designed to use TTA on group communication.

4.3.2 Data Aggregation for Multiple Groups

Data Aggregation for Multiple Groups (DAMiG) is designed to execute the Two-Tier Aggregation (TTA) on multiple group scenarios. To implement TTA, DAMiG computes an aggregation tree and applies statistical and merger aggregation functions along the path. DAMiG is able to reduce the energy consumption in scenarios with single or several groups. Moreover, the selection of internal and external group paths takes into account the residual energy of the nodes, avoiding the paths with low residual energy.

To execute internal and external group aggregation, DAMiG computes the following aggregation tree and paths:

- **Internal Group Aggregation Tree** (named IntraTree): It is the tree computed to aggregate the internal group traffic. The root of this tree is called Internal Group Sink (IGSink).
- **External Group Path** (ExPath): This is the path that connects two groups. In one end point, ExPath has the Internal Group Sink (IGSink), in the other end point, it has the External Group Receiver (EGReceiver). While the IGSink aggregates all internal group data, the EGReceiver receives data from other groups, aggregates it (using the payload concatenation functions) with its own data and forwards the aggregated data to the sink.
- **External Group Aggregation Tree** (called ExTree): It is the tree that connects all groups involved in aggregation. The root of the ExTree is the

network gateway.

Figure 4.4 shows the Internal Group Tree and the ExPath between two groups. In addition, it also depicts the nodes with special functions, IGSink and the EGReceiver. Considering the two groups, A sends its aggregated traffic to group B. The node in A that sends traffic to B is the IGSink and the node in B that receives the traffic from A is the EGReceiver.

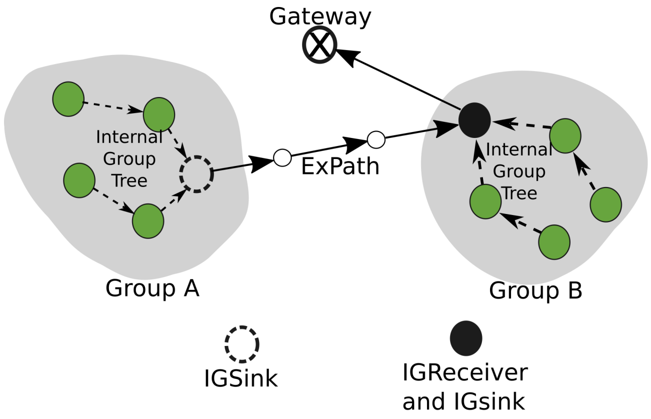


Figure 4.4: Internal and External-Group Communication.

DAMiG elects the pair (IGSink, EGReceiver) as the pair that minimizes the ratio between the Euclidean distance and the sum of residual energy, as presented in Equation 4.7. In this equation, the Euclidean distance between a pair of nodes (x, y) is computed by the function $distance(x, y)$, and the residual energy of a node x is given by the function $E_{res}(x)$.

$$\underset{\forall a \in A, \forall b \in B}{\arg\min} \left(\frac{distance(a, b)}{E_{res}(a) + E_{res}(b)} \right) \tag{4.7}$$

DAMiG uses Equation 4.7 because it is able to select a pair of nodes on the border of the groups that have more energy.

As groups can be located in any geographic area of the network, including locations out of the gateway range, DAMiG uses, if necessary, the communication capabilities of non-source nodes to forward data along the ExPath. To minimize the energy consumption, the ExPath is the shortest path between IGSink and EGReceiver.

The intraTree, the ExTree and the ExPaths use the following equation to estimate the cost of a link:

$$Cost(i, j) = \frac{Tx_i + Rx_j}{E_{res}(i) + E_{res}(j)} \tag{4.8}$$

Equation 4.8 estimates the cost for a node i to communicate with its neighbor j, where Tx_i and Rx_j (see Equation 4.1 and 4.2) denote the energy consumption to transmit and receive, respectively.

Algorithm 4.1 presents how DAMiG computes the ExTree and the IntraTrees. Firstly, it selects an IGSink and an EGReceiver for each pair of concurrent groups. This selection is based on Equation 4.7. Having the IGSink and EGReceiver, the proposed solution computes all possible ExPaths, which are the shortest paths between the IGSink and EGReceiver pairs. Then, the algorithm computes the ExTree, which is the Minimum Spanning Tree (MST) or Shortest Path Tree (SPT) of all ExPaths. Similar to the ExTree, all intraTree's are also computed based on MST or SPT.

Algorithm 4.1. Internal and External Group Aggregation Tree

```
Input  : $N(V, E), DataRequests, atvGroups$
Output: $IntraTree, ExTree$
Start
    for ($l \leftarrow 0; l < atvGroups.size; l++$) do
        for ($f \leftarrow l+1; f < atvGroups.size; f++$) do
            $IGSink \leftarrow selectIGSink(group[l]);$
            $EGReceiver \leftarrow selectEGRcv(group[f]);$
            $ExPath \leftarrow find(IGSink, EGReceiver);$
        end for
    end for
    $ExGroupTree \leftarrow findExTree(ExPath);$
    for ($j \leftarrow 0; j < actGroups.size; j++$) do
        $sink \leftarrow getIntraSink(ExGroupTree);$
        $intraGroupTree[j] \leftarrow findIntraTree(sink);$
    end for
End
```

The next section describes the second heuristic proposed for M2M group communication.

4.3.3 Two-Tier Aggregation for Multi-target Applications

This section describes the Two-Tier Aggregation for Multi-target Applications (TTAMA). First, in Section 4.3.3.1 there is an overview of the TTAMA operation. Following this, Section 4.3.3.2 shows how to schedule the groups and reduce the number of trees necessary to perform the Two Tier Aggregation over time. Next, Section 4.3.3.3 shows how to compute the Network Spanning Tree, which is a general aggregation tree. Finally, Section 4.3.3.4 presents how to find the TTAMA aggregation trees.

4.3.3.1 Overview

Two-Tier Aggregation for Multi-target Applications (TTAMA) aims to reduce the resources spent on data redundancies and minimize the message overhead. In the first aggregation tier, TTAMA executes aggregation functions to reduce data redundancy, and in the second, it applies an aggregation function to reduce the costs incurred by the message overhead.

TTAMA finds its routes in two phases. Firstly, TTAMA computes a Network Spanning Tree (NST), which includes all the network nodes. Then, it prunes the spanning tree to find a set of trees, called *TTAMA trees*, that will be able to route the external and Internal Group Traffic.

Figure 4.5 shows the TTAMA components and how they use the group settings to compute the aggregation trees. Figure 4.5 identifies the four main TTAMA components, which are described as follows:

1. The TTAMA component, called Group Settings Control, is responsible for storing and keeping information about the groups. For instance, this component stores the node IDs and the settings of each group. Figure 4.5 depicts that the Group Settings Control receives the communication settings from the Group Manager (GM). The GM controls the list of resources of the devices (e.g. sensors, actuators, and network settings), the data requests and executes the group management tasks, such as creation, deletion, and edition. Further details about the group formation procedures are provided by Liu et al. [Liu et al., 2014] and Ishaq et al. [Ishaq et al., 2014].
2. After receiving the information about the group settings, TTAMA triggers the algorithm that creates a Group Scheduler. This component mainly deals with two communication settings: communication periodicity and delay tolerance. Using these communication settings, the Group Scheduler computes the communication timings for each group. The aim of the Group Scheduler is to maximize the external group traffic, which increases the likelihood of external group aggregation. If the group settings do not change over the time, the Group Scheduler is only computed once.
3. The Network Spanning Tree (NST) is a component that does not depend on the Group Scheduler. The NST creates a tree that spans all the nodes that belong to at least one group. This component only needs to know the node IDs of each group, and this information is provided by the Group Settings Control.
4. Finally, the Set of TTAMA Trees (STT) uses the information provided by the Group Scheduler and the NST to compute a set of aggregation trees. Basically, the STT prunes the Network Spanning Tree (NST) for each different group combination present in the Group Scheduler. The STT output is a set of aggregation trees, which can be considered to be NST sub-trees. Each tree in STT will be used in accordance with the timings defined by the Group Scheduler.

After showing an overview of the TTAMA components, the following section shows more details about the Group Scheduler.

4.3.3.2 Group Scheduler

The aim of Group Scheduler is to find inside the delay tolerance interval a new communication periodicity that reduces the number of aggregation trees and

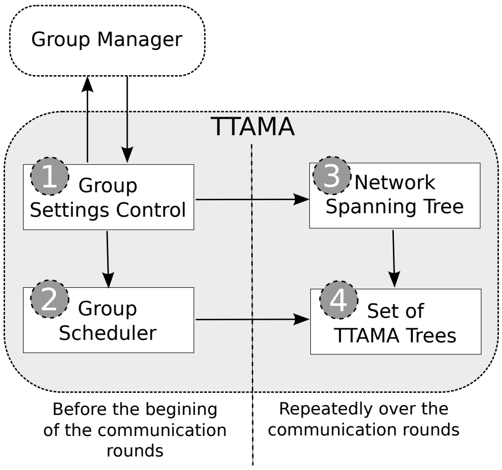


Figure 4.5: TTAMA operation.

increases the amount of external group traffic. The number of aggregation trees, is determined knowing the communication periodicity s^k_{freq} of each group.

The number of trees necessary to provide two tier aggregation grows exponentially with the number of groups with a single communication periodicity. The upper bound of groups is found when all the groups have different frequency s_{freq}. The lower bound is achieved when all the groups have the same communication periodicity. The upper bound results in the minimum amount of external group traffic, while the lower bound produces the maximum. Thus, the Group Scheduler uses the delay tolerance (i.e. s_{delay}) of the groups to reduce the number of trees, while maximizing the external group traffic.

The delay tolerance setting allows the Group Scheduler to delay the communication of a group, and find a future communication round which has a larger number of concurrent groups. In other words, the delay tolerance allows TTAMA to maximize the number of groups with equal communication periodicity. If it is not possible to equalize the communication periodicities, TTAMA attempts to maximize the number of concurrent groups in the same round.

This occurs, for instance, if the group A_1 is configured with communication periodicity 10 and the tolerable delay is 2. The group A_s is configured with communication periodicity 15 and the tolerable delay is 5. The number of aggregation trees necessary to provide two tier aggregation for A_1 and A_2 might be reduced if the periodicity communication of A_2 is delayed by 5 rounds (i.e. rounds that are a multiple of 20). After this delay, the communication periodicity of A_2 becomes divisible by A_1. This means that every communication round that A_2 needs to communicate its data readings, A_1 will communicate concurrently.

Summing up, the Group Scheduler influences two aspects of TTAMA. The first concerns the amount of external group traffic and consequently the execution of payload concatenation, which can be executed if there is at least two concurrent groups in a particular communication round. The second effect of Group Scheduler for TTAMA is the number of trees necessary to provide data aggregation during the communication rounds. Regarding both aspects, TTAMA finds communication periodicities inside the delay tolerance interval that increases the external group traffic and reduce the number trees.

Continuing to specify the TTAMA components, the next section details how the Network Spanning Trees are computed.

4.3.3.3 Network Spanning Tree

The Network Spanning Tree (NST) is the tree that includes all the nodes which belong to at least one group. As can be observed in Algorithm 4.2, first the NST computes a spanning tree for all the network nodes. After this, the NST prunes the leaf nodes that do not belong to any group. This pruning continues as long as there is a leaf node that is not a member of any group.

The pruning approach is preferable when there is a large number of nodes that belong to at least one group. Otherwise, the number of operations needed to prune the nodes could be high. It is important to notice that there are several well-known approaches to compute the NST. Although Algorithm 4.2 uses the Prim's algorithm [Prim, 1957], it is not limited to this algorithm.

Figure 4.6 illustrates Algorithm 4.2, and takes into account the existence of nodes that do not belong to any group. It can be observed in Figure 4.6.a that the initial NST includes all the network nodes. Thus, after the pruning process, the NST only includes the nodes that belong to at least one group.

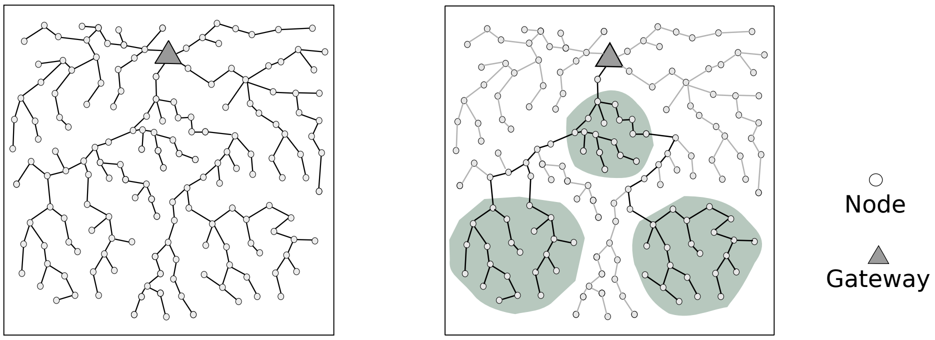
As the NST is based on Prim's algorithm, it finds the Minimum Spanning Tree of the edge-weighted graph $G(V, A)$. One important component of any edge-weighted solution is the cost function used to assign the weights to each edge. According to the variables defined in Section 4.3.1, the following equation gives the function used to assign a weight to each edge in the network.

$$Edge_{weight} = \frac{Tx_i + Rx_j}{E_{res}(i) + E_{res}(j)} \tag{4.9}$$

In Equation 4.9 the weight of an edge considers the energy necessary to communicate (i.e. Tx and Rx) and the residual energy in the transmitter and receiver. By using the residual energy as a denominator, this function assigns high weights

Algorithm 4.2. Network Spanning Tree

```
Input  : $N(V, E), mbrs[]$
Output: $NST$
Start
    $GwID \leftarrow 0$;
    $NST \leftarrow PrimAlgorithm(N(V, E), GwID)$;
    while ($NotMemberLeaf(NST, mbrs[]) \neq 0$) do
        $NST \leftarrow pruneNotMbrLeaf(NST, mbrs[])$;
    end while
End
```



(a) The initial Network Spanning Tree.

(b) The Network Spanning Tree after pruning the non member leaf nodes.

Figure 4.6: The Network Spanning Tree computation.

to the edges that have nodes with low residual energy. Avoiding these nodes is of crucial importance to extend the network lifetime, since they tend to deplete their energy in a short period of time.

The next section describes the aspects related to the TTAMA trees computation.

4.3.3.4 TTAMA Trees

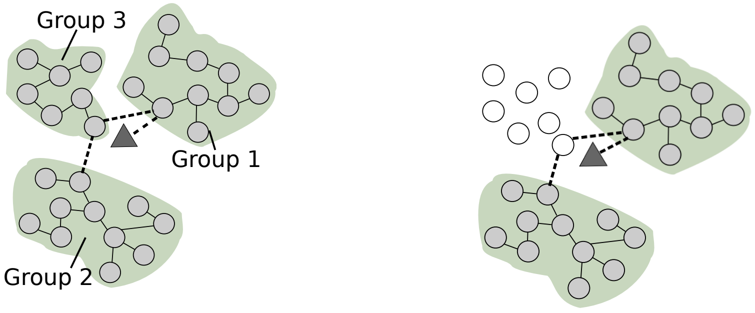
The computation of TTAMA trees is made using Algorithm 4.3. This Algorithm uses Group Scheduling and the NST to find the set of TTAMA trees. The first loop of the algorithm finds the group combinations present in the Group Scheduling. There should be one TTAMA tree for each Set of Concurrent Groups, which can be retrieved from the Group Scheduling.

Algorithm 4.3. TTAMA trees.

```
Input  : $groupScheduling[], NST$
Output: $ttamaTrees$
Start
    for ($k = 0; k < groupScheduling.Size; k++$) do
        $groups.Add(findConcurrentGroups(groupScheduling[k]))$;
    end for
    for ($i = 0; i < groups.Size(); i++$) do
        $memberNodes \leftarrow findMemberNodes(groups[i])$;
        $sTree \leftarrow NST$;
        while ($noMemberLeaf(sTree, memberNodes) \neq 0$) do
            $sTree \leftarrow pruneMemberLeaf(sTree)$;
        end while
        $ttamaTrees.Add(sTree)$;
    end for
End
```

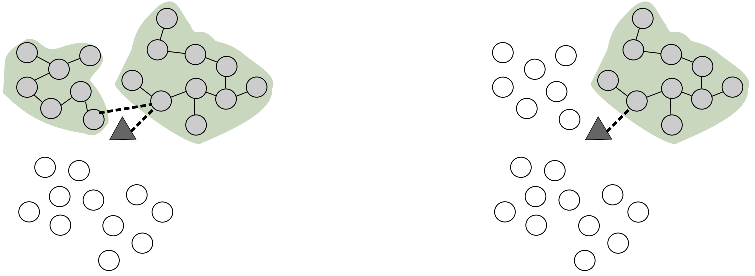
In the next loops, Algorithm 4.3 prunes from the NST all the nodes that are leaves and are not members of any group in the particular group combination.

For instance, assuming the existence of three groups that results in the following Group Scheduling: $\{A_1, A_2, A_3\}$, $\{A_1, A_2\}$, $\{A_1, A_3\}$ and $\{A_1\}$. Then, Algorithm 4.3 finds a TTAMA tree for each group combination. Figure 4.7 illustrates examples of TTAMA trees. Figure 4.7.a shows the TTAMA tree that is able to aggregate the traffic in the communication rounds when groups A_1, A_2, A_3 communicate concurrently. Similarly, the trees in Figures 4.7.b and 4.7.c aggregate the traffic of groups A_1, A_2 and A_1, A_3, respectively. Finally, the TTAMA tree in 4.7.d is used to aggregate the traffic of group A_1.



(a) TTAMA tree for groups A_1, A_2, A_3.

(b) TTAMA tree for groups A_1, A_2.



(c) TTAMA tree for groups A_1, A_3.

(d) TTAMA tree for group A_1.

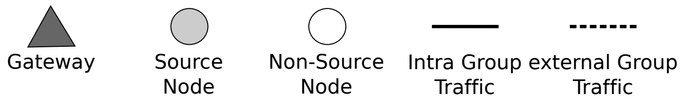


Figure 4.7: Example of TTAMA trees that are able to aggregate data of three groups.

After presenting the TTAMA components, the following section shows the performance evaluation.

4.3.4 Performance Evaluation and Results

The aim of this section is to evaluate the performance of TTAMA and DAMiG in comparison with related data aggregation approaches. Data Aggregation for Multiple Groups (DAMiG) is the first selected approach to compare with TTAMA. This solution performs the Two Tier Aggregation scheme using Minimum Spanning Tree (MST). Similarly, the second approach is a version of DAMiG that uses the Shortest Path Tree (SPT). The third solution is a version of DAMiG that builds an Internal Group Tree by selecting the parent with the highest level of residual energy. For clarification, the names of these solutions are DAMiG-MST, DAMiG-SPT and DAMiG-MaxEnergy, respectively.

In addition to the DAMiG versions, the fourth solution is an adapted version of the Energy Efficient Spanning tRee (EESR). The original EESR does not perform the Two Tier Aggregation approach. Hence, the adapted EESR forwards the Internal Group Traffic to the network gateway via the shortest path. It is worth noting that the original EESR energy cost metric was not changed, which means that the aggregation trees and the shortest path use the original EESR energy cost metric.

The main reason to compare TTAMA with DAMiG and EESR is to study the TTAMA performance enhancement when it is compared to solutions that perform two (i.e. DAMiG) or one-tier (i.e. EESR) of aggregation. Besides, the different versions of DAMiG allow to further apprimorate this comparision, since each DAMiG version computes the Internal Group Tree using different algorithms.

The folowing metrics were selected to measure the energy efficiency of the solutions:

- **Network lifetime**: The network lifetime is determined by counting the number of communication rounds until the first node depletes its energy [Karakus et al., 2013]. The number of communication rounds corresponds to the number of times all the active groups communicate their data to the network gateway.
- **Energy consumption**: It measures the energy consumed during a communication round based on Equation 4.4. It is important to notice that only the traffic going from the nodes to the gateway is taken into account for the energy consumption measurements. Other protocol messages (e.g. CoAP confirmation) have no impact in the energy resources.

The Constrained Application Protocol (CoAP) was used as reference to configure the traffic of the performance evaluation. The reason for this selection is that CoAP is the standard protocol for machine-to-machine communication and it has options, such as group communication and observe, enabling the exact type of traffic the solutions must be evaluated. It is considered that each group transmits one CoAP message, in accordance with the communication periodicities. Each CoAP message has a header of 12 bytes, which includes the fields: Version (2 bits), Type (2 bits), Token Length (4 bits), Code (8 bits), Message ID (16 bits), and Token (64 bits) [Shelby et al., 2014]. Moreover, the default CoAP payload size is 4 bytes. This payload is used to store the group ID and the data reading.

The communication periodicity of each group is determined by a uniform distribution. Moreover, the results were obtained on the basis of 40 samples of each setting with a confidence interval of 95%. Table 4.3 provides further information about the simulations.

Table 4.3: Configuration Parameters.

Parameter	**Value**
Header Size	12 Bytes
Default Payload Size of 1 message	4 Bytes
Maximum Transmission Unit	127 Bytes
Communication Periodicity	Random
Energy	1.8 - 2.2 J
Constant	e_{elec} 50x10^{-9}
Constant	e_{amp} 100x10^{-12}
Gateway Position	Center
Network Topology	Grid 140m x 150m
Network Size	100
Number of Groups	1, 2, 4, 5, 10, 20, 50, 100
Number of Group Members	100, 50, 25, 20, 10, 5, 2, 1
Wireless Range	30m

In order to evaluate the benefits and impact of the proposed schemes in a simulation environment, the solutions and the models described in Section 4.3.1 were implemented in Java. This simulation tool was conceived originally to evaluate the performance of DAMiG-MST and EERS in [Riker et al., 2014a]. For this evaluation, we also implemented DAMiG-SPT, DAMiG-MaxEnergy, and TTAMA. The simulation environment is a program which reads for every simulation an Extensible Markup Language (XML) configuration file, containing key parameters, such as initial energy, wireless range, number of devices, and group information. The output of the simulation is written in external files, containing the network lifetime, the energy consumption, and the residual energy. The validation of this tool was conducted for networks having 25, 50, and 100 nodes. The simulation tool and the implemented codes are available online[1].

The network topology is a grid of 140m x 150m with 100 nodes and 1 gateway (located in the center). The solutions were evaluated with the following number of concurrent CoAP groups: 1, 2, 4, 10, 25, 50, and 100. These numbers of groups resulted in the following number of members for each group: 100, 50, 20, 10, 4, 2, and 1. Figure 4.8 illustrates the number of nodes and the shape of each group.

Figure 4.9 shows the performance of TTAMA, DAMiG-MST, DAMiG-SPT, DAMiG-MaxEnergy, and Adapted EESR. TTAMA has a better performance for all of the CoAP groups. TTAMA has a longer network lifetime because the root node of the TTAMA trees might be located outside the group. This feature allows TTAMA to have a longer network lifetime when the groups have a large number of members. Another DAMiG feature that results in lower performance is the path computation for External Group Traffic. DAMiG creates a complete graph, where the edges correspond to the Shortest Path between every pair of

---

[1] `https://github.com/AggSimulationTool/v1.2`

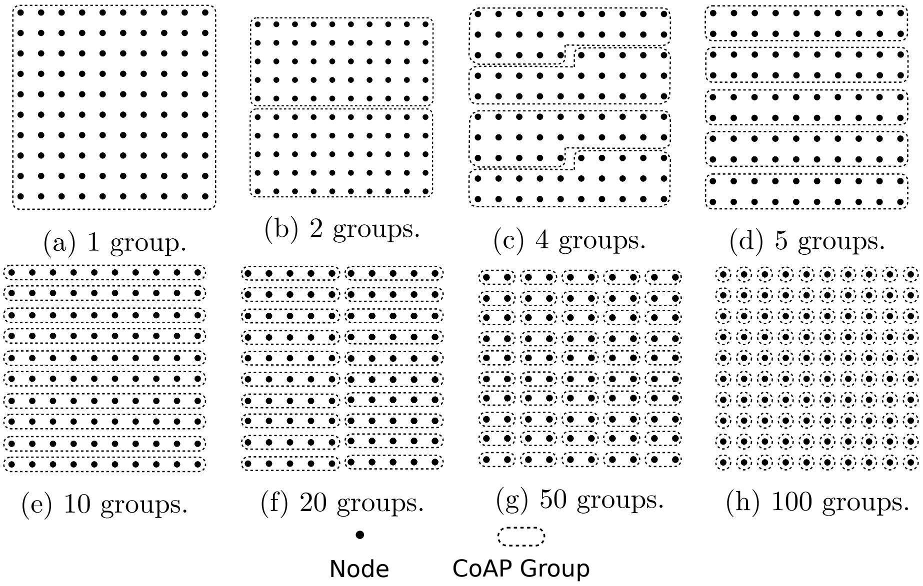

(a) 1 group. (b) 2 groups. (c) 4 groups. (d) 5 groups.
(e) 10 groups. (f) 20 groups. (g) 50 groups. (h) 100 groups.

Figure 4.8: The simulated CoAP groups in the network.

groups. This approach does not produce better results than those of TTAMA. For this reason, TTAMA approach has a superior performance to that of DAMiG and Adapted EESR.

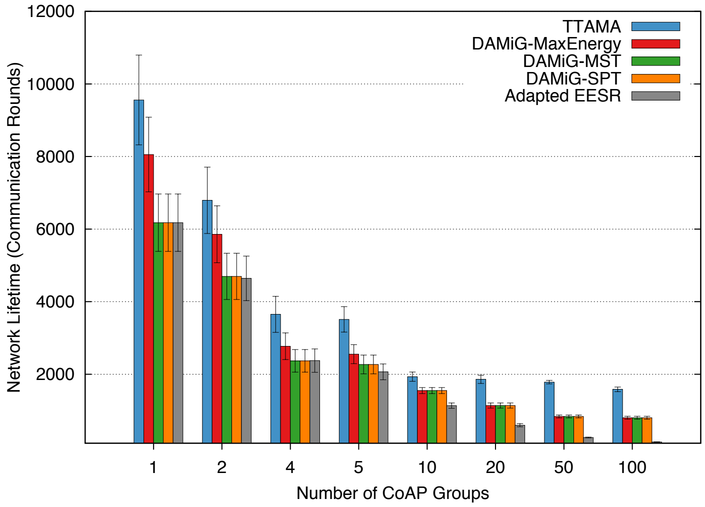


Figure 4.9: Network lifetime.

Based on data depicted in Figure 4.9, the existence of CoAP groups might seem disadvantageous, since with a single CoAP group is possible to obtain the longest network lifetime. However, with a single CoAP group, Payload Aggregation produces an overall data reporting. This aggregation will not allow multiple CoAP clients to obtain a fine-grained information from the received data reports (e.g. the average temperature in a specific region of the network).

Observing Figure 4.9, DAMiG-MST, DAMiG-SPT, and Adapted EESR have almost the same network lifetime when the number of CoAP groups is 1, 2 or

4. However, these DAMiG solutions have a better performance than Adapted EESR when the number of groups is greater than 4. The reason for this is that the Adapted EESR solution does not perform the Two Tier Aggregation scheme. Thus, EESR communicates more External Group Traffic than the TTAMA and DAMiG solutions.

In addition, among the three DAMiG versions, DAMiG-MaxEnergy is the solution with the longest network lifetime. This solution selects the parent with highest residual energy and is able to balance its energy consumption. However, the network lifetime of DAMiG-MaxEnergy is similar to DAMiG-MST and DAMiG-SPT when there is a decline in the members of the groups. For instance, in the case of one group with 100 members, the difference between DAMiG-MaxEnergy and the another DAMiG version is 23.32%. When there are five groups with 20 members, this difference is 11,12%.

To reduce the energy consumption during each communication round it is important to achieve longer network lifetime. Figure 4.10 shows the average energy consumption per communication round. As can be seen, TTAMA is the solution that spends least energy per communication round. On the other hand, Adapted EESR has the highest average energy consumption per communication round. Besides, the DAMiG versions and Adapted EESR tend to increase the energy consumption exponentially when the number of CoAP groups increases, since they poorly reduce the External Group Traffic overhead. For instance, when the number of CoAP groups changes from 1 to 5, TTAMA, DAMiG-MaxEnergy and Adapted EESR increase in 66%, 72% and 77% their energy consumption per communication round, respectively. Similarly, when the number of CoAP groups changes from 5 to 100, these values are 15%, 31% and 86%, respectively.

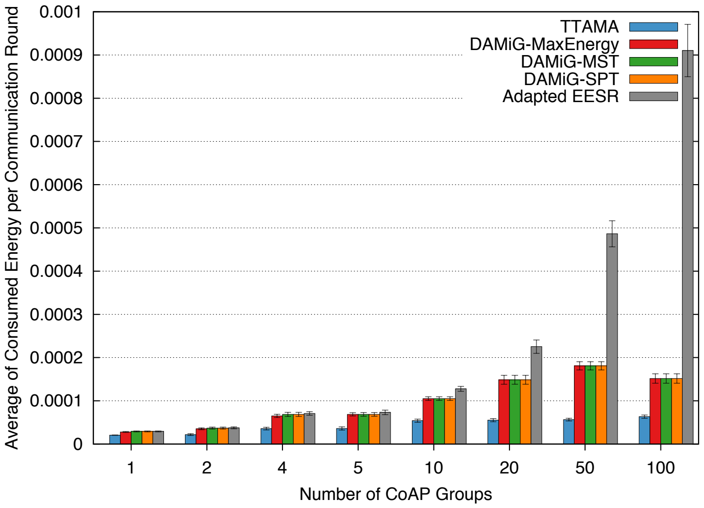
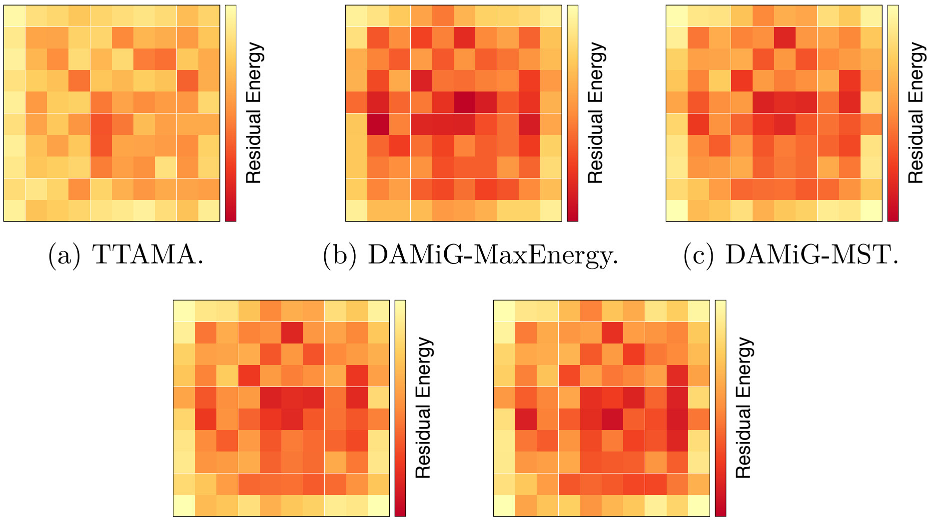


Figure 4.10: The energy consumption per communication round.

Figure 4.11 provides a visualization of the residual energy when the network lifetime ends. In this figure, the red color means minimum and the white color means maximum residual energy. Thus, Figure 4.11 reveals that there is a



(a) TTAMA. (b) DAMiG-MaxEnergy. (c) DAMiG-MST.

(d) DAMiG-SPT. (e) Adapted EESR.

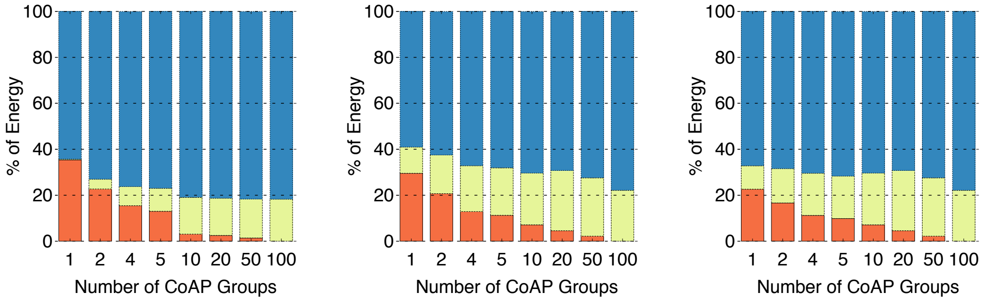
Figure 4.11: The color map of the residual energy.

noticeable difference between TTAMA and the other solutions. TTAMA has a larger amount of residual energy than the Adapter EESR and the DAMiG versions.

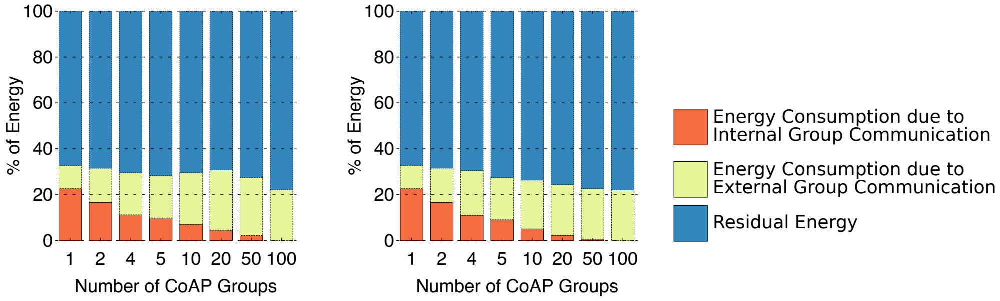
While Figure 4.11 is useful for visualization purposes, Figure 4.12 shows more details about the average residual and consumed energy (i.e. Internal Group and External Group energy) when the first node depletes its energy. On one hand, it can be seen that the External Group communication consumes more energy when the number of CoAP groups increases. The reason for this is that the External Group Traffic grows with the number of groups. On the other hand, there is a reduction in consumption by the Internal Group when there is a growth in the number of groups.

Two main aspects contribute to the shorter network lifetime and higher residual energy when the number of groups increases. Payload Aggregation reduces its data more effectively than Payload Concatenation, since the Concatenation of payloads reduces only the extra headers of the External Group Traffic messages and Payload Aggregation are executed on the payloads of the Internal Group Traffic. This means that the consumed energy per communication round is lower when the solutions use Payload Aggregation rather than Concatenation.

On the basis of the data in Figures 4.9 and 4.12, it can be noted that TTAMA has the best performance in terms of network lifetime and is the solution with the lowest energy consumption. The large amount of residual energy shown in Figure 4.12 can be explained by the well-known phenomenon called the Energy Hole Problem [Kleerekoper and Filer, 2014], which occurs when the nodes in the region near the Gateway deplete their energy before the other nodes. Although the network has, on average, a large amount of residual energy, due to the energy hole problem, the nodes near the Gateway have a lack of residual energy.



(a) TTAMA. (b) DAMiG-MaxEnergy. (c) DAMiG-MST.



(d) DAMiG-SPT. (e) Adapted EESR.

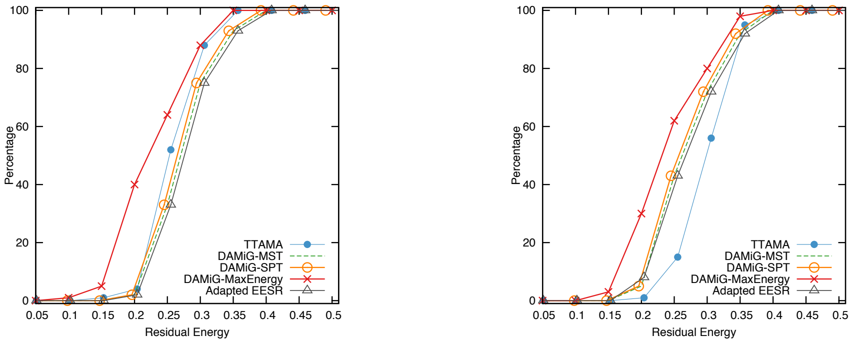
Figure 4.12: The average percent of External, Internal, and Residual Energy.

Observing the data presented by Figure 4.12, it is possible to question the need of existence of the Internal Group Traffic aggregation. The argument for that might be based on the fact that the energy consumed is lowest for all solutions if only the External Group aggregation is executed, for instance 100 CoAP groups. However, it is worth to observe that if only the External Group aggregation is executed the network lifetime is the lowest for all solutions (see Figure 4.9). Therefore, the Internal Group Traffic aggregation contributes to the energy consumption, but it extends the network lifetime.

Figure 4.13 shows the probability distribution function of residual energy when the first node depletes its energy. On the basis of the data supplied by Figure 4.13 it is possible to find out the percentage of nodes that have at least a certain amount of residual energy. Figures 4.13.a, 4.13.b, and 4.13.c show that DAMiG-MaxEnergy is the solution with the highest percentage of nodes with low residual energy when the number of CoAP groups is 1, 2, and 4. This behavior changes when the number of CoAP groups is 5, 10, 20, 50 and 100, since DAMiG-MaxEnergy and the other DAMiG versions have almost the same values.

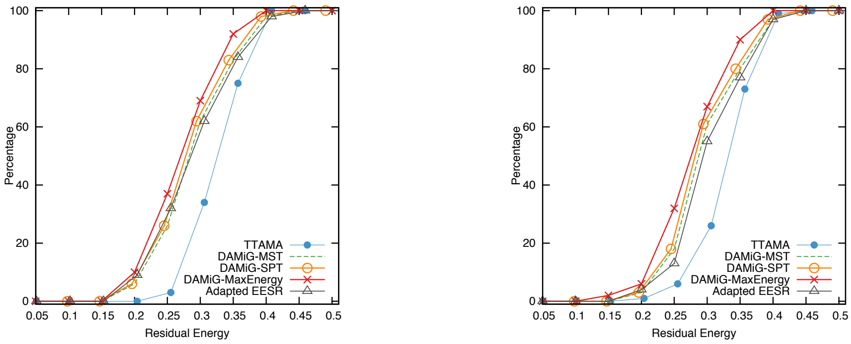
Moreover, Figure 4.13 shows that TTAMA has the lowest percentage of nodes with low residual energy, except for the case of 1 CoAP group. In some cases (e.g. 10, 20, and 50 CoAP groups), the Adapted EESR is the solution that approximates most closely to TTAMA. However, the performance of Adapted EESR in terms of network lifetime is worse than TTAMA. Thus, the good values achieved by Adapted EESR in terms of residual energy are not reflected in the primary goal, which is the network lifetime.

The next section presents the optimization model developed to find the max-



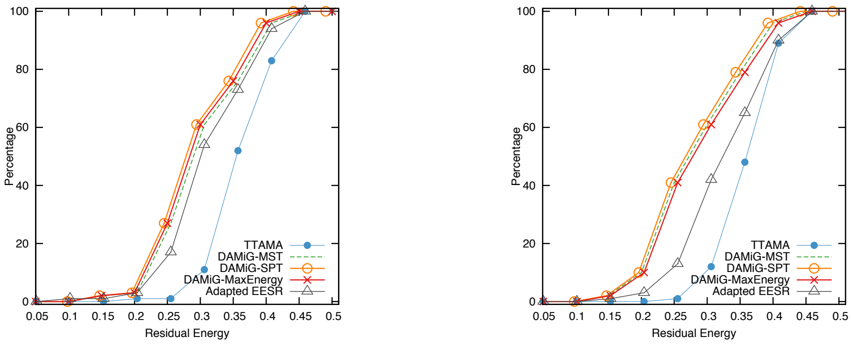
(a) 1 CoAP Groups.

(b) 2 CoAP Groups.



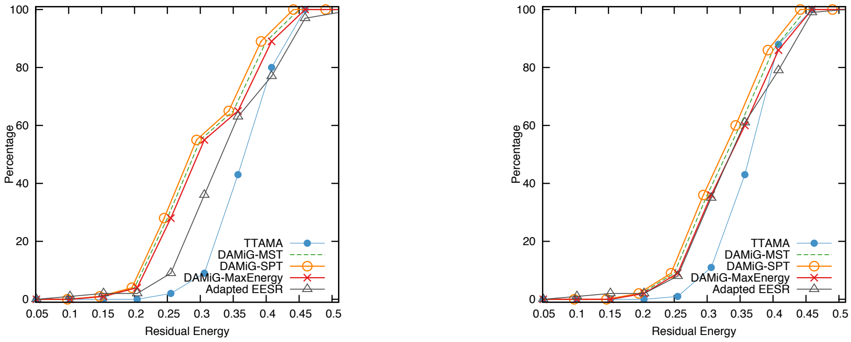
(c) 4 CoAP Groups.

(d) 5 CoAP Groups.



(e) 10 CoAP Groups.

(f) 20 CoAP Groups.



(g) 50 CoAP Groups.

(h) 100 CoAP Groups.

Figure 4.13: The probability distribution function of the residual energy when the first node depletes its energy.

imum network lifetime of Two Tier Aggregation.

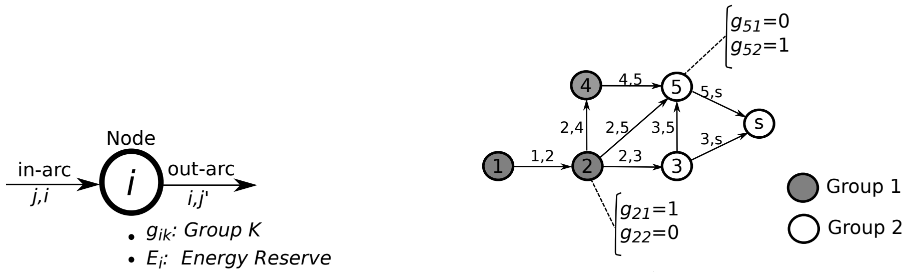
4.4 Network Lifetime Upper Bound

This section presents the mathematical model to optimize the Two Tier Aggregation. The main objective of the proposed optimization model is to obtain an upper bound of the network lifetime and compare it with different state-of-the-art heuristic solutions.

The Two Tier Aggregation (TTA) approach inserts many payloads in one message to efficiently gather data from multiple groups. The current heuristics for the TTA do not guarantee the optimal results in terms of network lifetime, in many cases they can perform poorly. Therefore, a model that optimizes the network lifetime of TTA is very important to provide a performance benchmark for the heuristics. Given the fact formulation is centralized, which demands information about the network status, it is not recommended to be implemented in real networks. Nevertheless, the proposed model is still a valuable approach for analysis and comparison performance.

The optimization of TTA is formulated as a lexicographic multi-objective Integer Linear Program (ILP) that is able to find the network flows that achieve the upper bound of the network lifetime. This formulation is proposed considering single and multiple sinks on the network.

This section is structured as followed. Sections 4.4.1 presents the network model and the problem statement. Sections 4.4.2 and 4.4.3 present the model formulation considering single and multiple sinks, respectively. Section 4.4.4 describes the obtained results.



a) Node i of the Graph.

b) Example.

Figure 4.14: Ilustration of the Graph Model.

4.4.1 Network Model and Problem Statement

The network is modeled as a directed graph $G(V, A)$, where $V = \{1, 2, ..., n\}$ is the set of nodes and $A = \{(i, j)|i, j \in V\}$ is the set of arcs. The network has a single sink node, denoted by $s \in V$, which is the final destination of all messages. For simplicity, the set of regular nodes (i.e. non-sink nodes) is denoted as $N = V - \{s\}$. In addition, each node i has an energy reserve denoted by E_i, and the monitoring groups are denoted by $S = \{1, 2, ..., k\}$. The following binary values indicate the group that a node i belongs to:

$$g_{ik} = \begin{cases} 1 & \text{if node } i \text{ belongs to group } k \\ 0 & \text{otherwise} \end{cases}$$

Figure 4.14 illustrates the graph model used. Figure 4.14.a shows the arcs of a node i, and Figure 4.14.b presents an example of a graph with 5 nodes and a sink. As can be observed, nodes *1, 2, 4* belong to group 1, while nodes *3* and *5* belong to group 2.

According to Equation 4.5, $NL = min_{\forall i \in N} L_i$, the problem of maximizing the lifetime can be named as the *Max-Lifetime TTA Problem*. From the mathematical perspective, the Max-Lifetime TTA Problem is solved by the set of network flows that achieves the maximum network lifetime and does not violate the rules of payload aggregation and concatenation defined by the Two Tier Aggregation approach. This problem can be solved by maximizing the minimum L_i. Given that all produced traffic has to be communicated over the network, the solution for the Max-Lifetime TTA Problem is the network flow that maximizes the payload production (i.e. NL).

The following section outlines the formulation of the mathematical model.

4.4.2 Integer Linear Programming Formulation

The number of k-group payloads traveling over the arc *(i,j)* is represented by p_{ijk}. Besides, as it is not necessary to identify the headers in terms of groups, the number of headers traveling over an arc *(i,j)* is denoted by h_{ij}.

Setting a default size for payloads and headers, it is possible to define *Htx* and *Ptx* as the transmission energy for a node i to transmit a single header and a single payload, respectively. Similarly, *Hrx* and *Prx* can be defined as the energy needed for node i to receive a single header and a single payload on arc *(i,j)*, respectively. Table 4.4 summarizes the symbols related to the optimization model.

Table 4.4: Definition of terms.

Symbol	**Definition**
E_i	Initial energy of node $i \in N$.
p_{ijk}	Number of k-group payloads travelling over the arc (i,j).
h_{ij}	Number of headers travelling over an arc (i,j).
Htx	Energy consumed to transmit a single header.
Ptx	Energy consumed to transmit a single payload.
Hrx	Energy consumed to receive a single header.
Prx	Energy consumed to receive a single payload.
Tx_{setup}	Energy spent related to the CSMA and CCA procedures.
E_{slp}	Energy consumed per communication round in sleeping state.
E_{CPU}	Energy spent by the CPU to perform aggregation on a single payload.

Equation 4.10 defines the energy consumed by a node $i \in N$ regarding the communication of data. It is important to notice that Equation 4.10 also includes Tx_{setup} as the energy consumed to setup wireless transmissions. Tx_{setup} captures the energy consumption involved in the tasks before the transmission, which include the energy to perform Carrier Sense Multiple Access (CSMA) back off, Channel Clear Assessment (CCA) detection, and the change from Rx (CCA) to Tx mode. Section 4.4.4 further describes the values assigned to all parameters used in the energy model equations, including E_{CPU}, Tx_{setup}, *Htx*, *Hrx*, *Ptx*, *Prx*, and E_{slp}.

$$Ecomm_i = \sum_{j:(i,j)\in A} h_{ij}(Tx_{setup} + Htx) + \sum_{j:(i,j)\in A}\sum_{k\in S} p_{ijk}Ptx + \sum_{j:(j,i)\in A} h_{ji}Hrx + \sum_{j:(j,i)\in A}\sum_{k\in S} p_{jik}Prx \tag{4.10}$$

A node also consumes energy performing CPU operations to aggregate data, as presented in Equation 4.2.

$$Eagg_i = \sum_{j:(j,i)\in A} \sum_{k\in S} p_{jik} g_{ik} E_{CPU} \tag{4.2}$$

Header aggregation is not included in Equation 4.2, since all headers received by a node i are dropped. The total energy spent of node i on sleeping mode is $Esleep_i$, and it is computed as follows:

$$Esleep_i = NL\ E_{slp} \tag{4.3}$$

Here, E_{slp} is a term that represents the amount of energy per communication round spent on keeping the CPU in low power mode and the transceiver is turned off. In Equation 4.3, E_{slp} is multiplied by NL because it is the number of communication rounds.

Finally, the total energy consumption of a node $i \in N$ is defined in Equation 4.4.

$$Nspent_i = \overbrace{Ecomm_i}^{\text{Communication}} + \overbrace{Eagg_i}^{\text{Aggregegation}} + \overbrace{Esleep_i}^{\text{Sleep mode}} \tag{4.4}$$

Knowing the particular energy consumption of each node (given in Equation 4.4), ES denotes the energy spent by the network, and is defined in Equation 4.5.

$$ES = \sum_{i=1}^{n} Nspent_i \tag{4.5}$$

Besides, it is necessary for the ILP formulation to define a term, λ, that produces values less than 1 when it is multiplied by ES. Thus, λ is defined as:

$$\lambda = \frac{1}{1 + \sum_{i=1}^{n} E_i} \tag{4.6}$$

Lifetime-optimal solutions might differ with respect to network energy consumed, given by ES.

Considering two lifetime-optimal solutions, S_1^* and S_2^*, the network energy consumed resulting from S_1^* might be higher than S_2^*. Thus, $\max NL$ does not guarantee that the solutions found are the ones that consume the minimum network energy among all lifetime-optimal solutions. In order to guarantee both objectives, we formulate this problem using a lexicographic multi-objective approach. The primary objective is to maximize NL, while the secondary objective is to minimize the network energy consumption. The secondary objective can be achieved by setting λES as penalty term in the objective function. This term will be a secondary objective because it will always be less than 1, while NL is an integer greater than or equal to 1. Therefore, defining the objective as $\max\ NL - \lambda ES$, NL becomes preferred over λES.

Equation 4.7 defines the multi-objective function as the network lifetime, to be maximized, penalized by λES, leading to a formulation that minimizes the network energy consumption among all lifetime-optimal configurations. The constraints in Equation 4.8 model the fact that the total energy consumed by a node i must be less than or equal to its energy reserve E_i.

$$\max\ NL - \lambda ES \tag{4.7}$$

Subject to:

$$Nspent_i \leq E_i, \quad \forall i \in N \tag{4.8}$$

$$\sum_{j:(i,j)\in A} p_{ijk} = NL, \quad \forall i \in N, \quad \forall k \in S : g_{ik} = 1 \tag{4.9}$$

$$h_{ij} \geq p_{ijk}, \quad \forall (i,j) \in A, \quad \forall k \in S \tag{4.10}$$

$$\sum_{j:(i,j)\in A} \sum_{k\in S} p_{ijk}(1 - g_{ik}) = \sum_{j:(j,i)\in A} \sum_{k\in S} p_{jik}(1 - g_{ik}), \quad \forall i \in N \tag{4.11}$$

$$\sum_{u:(u,s)\in A} p_{usk} \geq NL, \quad \forall k \in S \tag{4.12}$$

$$p_{ijk} = 0, \quad \forall i,j \in N, \quad \forall k \in S : g_{jk} = 1 \land g_{ik} = 0 \tag{4.13}$$

$$NL \in \mathbb{Z}^+, \tag{4.14}$$

$$h_{ij}, p_{ijk} \in \mathbb{Z}^+, \quad \forall (i,j) \in A, \quad \forall k \in S \tag{4.15}$$

The constraints in Equation 4.9 are related to the payload aggregation executed on Internal Group Traffic. These constraints ensure that each node i produces, in the course of its lifetime, all payloads belonging to group k (to which node i belongs) that travel over the arcs originating in that node. This means that p_{ijk} is not affected by the number of payloads of the same group arriving from other nodes, since payload aggregation functions produce a single output value, regardless of the number of inputs.

It occurs because each numeric value taken as input of the payload aggregation function is defined as a primitive datatype, so both the input and the output values have a fixed number of bits. For a given set of input numbers defined as 32-bit Float, the payload aggregation function will produce a single 32-bit Float number as result.

The constraints in Equation 4.10 state that the number of headers travelling on an arc $(i,j) \in A$, which is given by h_{ij}, must be greater than or equal to the greatest payload of group k travelling on this arc, i.e. p_{ijk}. It is important to note that these constraints do not ensure that the number of headers is minimal. However, due to the energy consumption penalty term in the objective function, the number of headers in the final solution will be as small as possible for the corresponding number of produced payloads.

The constraints in Equation 4.11 enforce that each node i transmits to its neighbors all the payloads received by i that were produced in another group (i.e. when $g_{ik} = 0$). Hence, these constraints conserve the payloads produced by external groups, avoiding that non-member nodes perform payload aggregation.

The constraints in Equation 4.12 ensure a minimum number of payloads delivered to the sink node. These constraints impose that the number of k-group payloads delivered to the sink is greater than or equal to *NL*, which would correspond to the maximum payload aggregation. Besides, the constraints in Equation 4.13 exist to avoid payload loops over the groups. These constraints state that a k-group node j (i.e. $g_{jk} = 1$) cannot receive k-group payloads from a node i that does not belong to k (i.e. $g_{ik} = 0$). Finally, the constraints in Equation 4.14 and 4.15 require that the variables *NL*, p_{ijk} and h_{ij} are all integers greater than or equal to zero.

It is important to notice that the literature classifies a Linear Program as Integer Linear Program when all decision variables must be integers [Bradley et al., 1977]. If some, but not all, variables are restricted to be integer, it is called Mixed Integer Linear Program. In the case of our formulation, all decision variables are integer, therefore it is called Integer Linear Program. In some parts, the proposed formulation is not restricted to be integer (e.g. E_i and $Nspent_i$), but the decision variables continue to be integer.

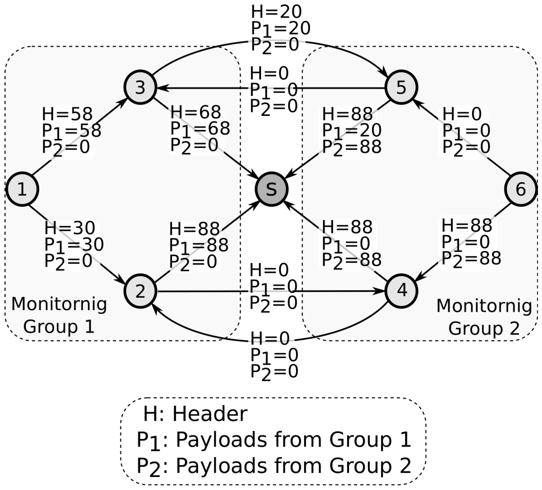


Figure 4.15: ILP solution example.

Figure 4.15 illustrates an optimal solution obtained from the proposed ILP, considering a network composed of six nodes and one sink. The nodes apply Payload Aggregation on the payloads of the Internal Group Traffic and remove the redundant headers from the messages originating in external groups. The existence of two groups creates three commodities in the network, namely payloads originating in groups 1 and 2, and headers. Nodes 1 and 6 do not communicate with external groups, while nodes 2, 3, 4, and 5 perform external group communication. Taking node 5 as example, it receives 20 payloads originating in group 1 from node 3, produces 88 payloads, communicating a total of 108 payloads and 88 headers. It means that node 5 assembled 20 messages with 2 payloads.

The next section introduces the necessary changes in the model to consider the case of multiple sinks.

4.4.3 Multiple Sink Formulation

Networks having multiple sinks are an important scenario, especially for large scale monitoring periodic applications. One of the advantages of multiple sinks is that the network has more than one connection with the core network, making the data communication more reliable. Besides, multiple sinks mitigate the communication bottleneck that occurs when there is a single sink over the network.

To optimize the network lifetime using TTA in multiple sinks networks, the optimization model should consider a set of sinks as final destination, which is denoted as $S = \{s_1, s_2, ..., s_w\}$. Thus, a messages going out a particular node has multiple sinks options to be sent, but only one sink is selected as destination, which means that the same message is not received by multiple sinks. Knowing this, the new version of the Equation 11 should consider all $s_m \in S$, as it is presented in Equation 4.16:

$$\sum_{s_m \in S} \sum_{u:(u,s_m) \in A} p_{us_mk} \geq NL, \; \forall k \in S \tag{4.16}$$

This new constraint states that payloads delivered to all sinks will count as valid payload communication. Special attention should be given to this constraint since it can be easily misunderstood. Equation 4.16 ensures that the number of payloads delivered to any sink must be equal or greater than the *NL*. As *NL* = min L_i, the constraints in Equation 4.16 ensure that the sinks cannot receive less than the min L_i, since min L_i is the least number of produced payloads. For instance, the constraints in Equation 4.16 force the sinks to receive at least one payload from each group k in a communication round *c*. In some cases, depending on the layout and the size of the group *k*, the sinks will receive more than one payload from group *k* in a communication round *c*.

This constraint also influences the level of the payload aggregation performed on the Internal Group Traffic. It is worth to observe that Equation 4.16 limits the *maximum* aggregation level allowed on the Internal Group Traffic. On the opposite side, the case of minimum aggregation level occurs when the set of sinks receives all payloads produced by the members of the group, which means payload aggregation does not take place.

The section below presents the evaluation and the obtained results.

4.4.4 Performance Evaluation and Results

This section shows the performance of the proposed ILP model measured in different network and group sizes. Section 4.4.4.1 gives details about the values assigned to the parameters of the ILP model. Section 4.4.4.2 presents the group and network sizes and layout. Section 4.4.4.3 shows the network lifetime and the energy consumption of the proposed ILP, and also presents a comparison between the upper bound obtained by the ILP and the heuristics taken from literature, namely Data Aggregation for Multiple Groups (DAMiG) [Riker et al., 2014a], Two Tier Aggregation for Multi-target Applications (TTAMA) [Riker et al., 2016a], and Energy Efficient Spanning tRee (EESR) [Hussain and Islam, 2007].

4.4.4.1 Parameter Configuration

The values assigned to the parameters of the proposed ILP are based on the ATMega128L [Atmel, 2016] micro-controller and the CC2420 transceiver. Based on IEEE 802.15.4 [IEEE, 2016], the data rate communication defined as 250 Kbps, and the communication round periodicity is 10s. Most of the non-decision variables are shown in Table 4.5. At 250 Kbps, the time necessary to communicate (i.e. Tx or Rx) a header of 48 Bytes is 0.037ms. Similarly, the time to communicate (i.e. Tx or Rx) a payload of 4 bytes is 0.012ms. The proposed solution is not restricted to a specific technology or protocol. However, reference standards are used to define the values of the parameters. CoAP [Shelby et al., 2014] has been used as reference to define the size of the payload. Regarding the size of the header, we considered a protocol stack composed of CoAP, UDP, 6LowPAN, and IEEE 802.15.4.

According to Casilari et al. [Casilari et al., 2010], before each transmission, a node has to perform setup procedures, which include the periods of Carrier Sense Multiple Access (CSMA) back off (2.24ms), Channel Clear Assessment (CCA) detection (0.128ms), and turnaround from Rx to Tx mode (0.192ms). Thus, Tx_{setup} is 2.56ms x 17.4mA x 3v= 0.134 mJ.

Table 4.5: Values for the Energy Consumption Model.

Symbol	**Energy Value**	**Duration**	**Current**
Htx	0.6438μJ	0.037 ms	17.4 mA
Tx_{setup}	0.134mJ	2.56ms	17.4 mA
Hrx	0.6956μJ	0.037 ms	18.8 mA
E_{CPU}	1.24μJ	51.7 μs	5 mA
Ptx	0.2088μJ	0.012 ms	17.4 mA
E_{slp}	–	–	15 μA
Prx	0.2256μJ	0.012 ms	18.8 mA
E_i	10J	–	–

Default Voltage is 3V.

Regarding the computation costs, each cycle of ATMega128L lasts 1.25e^{-7}s. The Application Report[2] shows that the ATMega128L executes on average 414 cycles to compute one simple math operation (i.e. addition, subtraction, multiplication, or division) on two 32-bit float numbers. As the aggregation functions considered involve also simple math operations, we consider 414 as the number of cycles necessary to aggregate two data payloads. This number of cycles lasts 1.25e^{-7} x 414 = 51.7μs. Thus, E_{CPU} corresponds to 51.7μs x 5mA x 3v = 1.24 μJ. For all nodes, the energy reserve E_i corresponds to 10 J. Finally, the time in

---

[2] www.ti.com/lit/an/slaa205c/slaa205c.pdf

sleeping mode per communication round corresponds to the time in which the node is not in any other state.

The next section describes the aspects related to the groups over the network.

4.4.4.2 Layout and Size of Groups

The size and layout of the network and groups have a strong influence on the energy consumption and on the network lifetime. In order to evaluate a representative number of scenarios, the evaluation has followed two approaches regarding the layout, size and number of groups.

- **Constant network size with variable number of group members (Constant Network Size):** In this approach, a network size is defined and the groups with equal number of members are distributed over the network. Seeking the fairness between the groups, all groups should have the same number of members, which means that it is possible to find the group size by finding the natural divisors of the network size. For instance, in the case of a network having 100 nodes, the group sizes that will form groups with the same number of members are: 1, 2, 4, 5, 10, 20, 25, 50, and 100.
- **Variable network sizes with constant number of group members (Variable Network Size):** In this approach, the number of members for each group is constant, while the number of groups and network size are variable. In this case, the fairness between the groups is also kept, since all groups should have the same number of members. For instance, defining the size of the groups as 4 and the total number of groups as 10, the network size will be 40. To obtain a ”square” layout, the groups’ sizes are given by power of two (e.g. 2x2, 3x3, and 4x4).

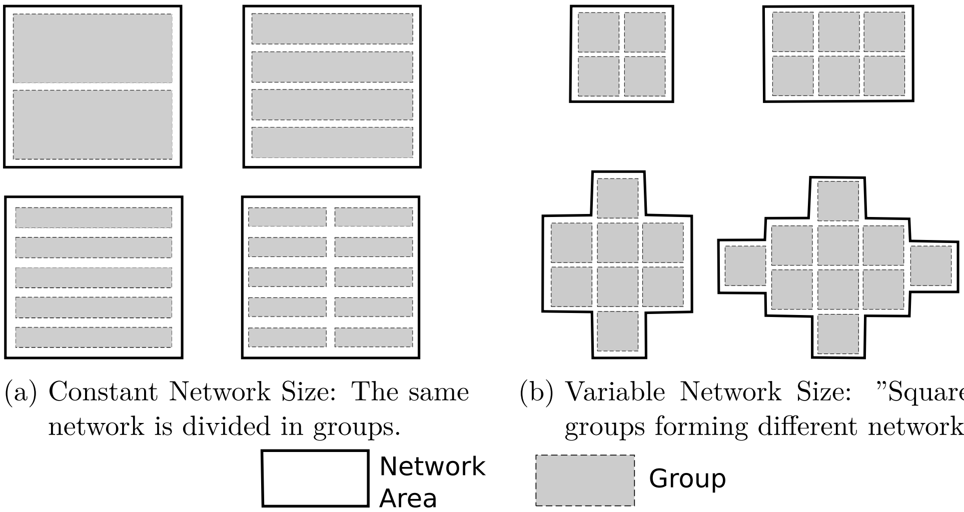
Figure 4.16 illustrates the two approaches applied to form the evaluated networks and groups.

All the evaluated nodes have a wireless range of 20m, while the horizontal and vertical spacing between the nodes is 20m, so the nodes have a maximum of 4 neighbors.

The following section presents and discusses the obtained results in terms of network lifetime and energy consumption.

4.4.4.3 ILP Network Lifetime and Energy Consumption

The main performance metric used in this evaluation is Network Lifetime (see Def. 3). Besides, as the energy consumption is a secondary objective, it is also used as a performance metric.



(a) Constant Network Size: The same network is divided in groups.

(b) Variable Network Size: ”Square” groups forming different networks.

Figure 4.16: Groups Distribution.

Figure 4.17 shows the network lifetime obtained by the proposed ILP considering a variable network size with constant number of group members (see Figure 4.16.b). The network lifetime is presented for networks having 4, 6, 8, and 10 groups.

As can be seen, the number of groups over the network impacts the network lifetime, since it controls the amount of External Group Traffic. However, the network lifetime is not necessarily impacted when the number of members of each group increases. For instance, considering a network with 10 groups, the obtained network lifetime is almost constant when the group size is 25, 36, 49, and 64 nodes.

These results (Figure 4.17) indicate that for a particular number of groups, the ILP model reaches a stable lifetime even if the number of group members increases. The reason for the lifetime stability is that the number of payloads

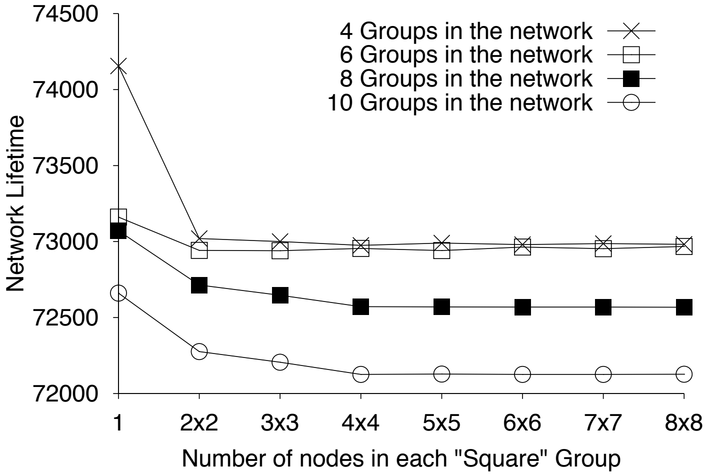


Figure 4.17: ILP Network Lifetime (Variable Network Size).

going outside a group is reduced to one regardless the group size.

Considering a constant network size with variable number of group members (See Figure 4.16.a), Figure 4.18 shows the network lifetime and the total network energy consumption due to the transmission of headers, internal group payloads, and external group payloads. There are fewer internal group payloads when the groups become smaller, causing a decrease in energy consumption. Due to External Group Traffic, the energy consumption for 50 monitoring groups is the highest, and the lifetime is the lowest.

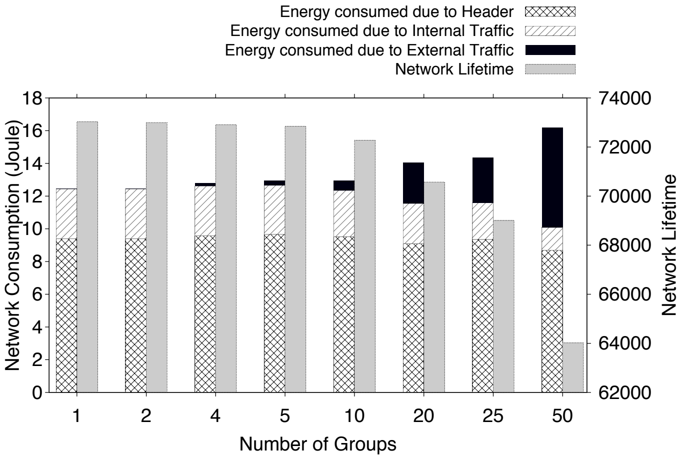


Figure 4.18: ILP Lifetime and Energy Consumption (Constant Network Size - 100 nodes).

Another relevant aspect is the aggregation behavior of the optimal solution obtained by the ILP. In this direction, Table 4.6 shows for different number of groups the average number of produced, delivered and aggregated payloads.

The percentage of payload aggregation in Table 4.6 is the ratio between the sum of payloads delivered to the sink and the sum of payloads produced. The maximum Internal Group Aggregation that a group can achieve is limited by the number of members. For example, when a network of 100 nodes has 50 groups, the maximum Internal Group Aggregation is 50%, since there are 2 members in each group and at least 1 payload should be communicated to the sink. For the case of 20 groups, each group has 5 members, which means that the maximum Internal Group Aggregation is 4/5= 80%. It is possible to notice that when the network has 1 and 2 groups, the Internal Group Aggregation is around 95%, which is not the maximum Internal Group Aggregation. However, when the network has 4, 5, 10, 20, or 50 groups, the percentage of aggregation decreases, achieving the maximum allowed Internal Group Aggregation. In our evaluation, TTAMA shows a similar level of payload aggregation, but it does not achieve the same results of the ILP when there are many groups. This indicates that TTAMA uses a correct approach to perform payload aggregation, but fails to find the paths to perform efficient payload concatenation.

Another interesting point of evaluation is the comparison between the network

Table 4.6: ILP Payload Statistics
(Constant Network Size - 100 nodes).

Number	**Payloads**			
of Groups	**Produced**	**Delivered**	**Aggregated***	**Aggregation**
1	7303000	292120	7010880	96 %
2	3651400	146056	3505344	96 %
4	1822650	72910	1749740	96 %
5	1456840	72944	1383896	95 %
10	722740	72184	650556	90 %
20	352850	70492	282358	80 %
50	128040	64466	63574	50 %

*These payloads do not go outside the group.

Table 4.7: ILP and Heuristics Comparison (Constant Network Size - 100 nodes).

	Number of Groups							
	1	**2**	**4**	**5**	**10**	**20**	**25**	**50**
ILP Network Lifetime	73030	73028	72906	72842	72274	70570	69006	64020
TTAMA's Network Lifetime	72190	67030	55770	51160	39830	40060	38520	38204
TTAMA and ILP Ratio	98.85%	91.79%	76.50%	70.23%	55.11%	56.77%	55.82%	59.68%
DAMiG's Network Lifetime	36590	24570	17660	16280	13250	11340	10980	7870
DAMiG and ILP Ratio	50.1%	33.6%	24.22%	22.35%	18.33%	16.07%	15.91%	12.29%
EERS's Network Lifetime	36590	24610	14740	12270	7020	3660	3250	1550
EERS and ILP Ratio	50.1%	33.7%	20.22%	16.84%	9.71%	5.19%	4.71%	2.42%

lifetime obtained from the ILP and the state-of-art heuristics, namely DAMiG [Riker et al., 2014a], TTAMA [Riker et al., 2016a], and EESR [Hussain and Islam, 2007]. As Table 4.7 presents, the proposed ILP solution has the best performance, which is expected because it achieves the network lifetime upper bound of TTA. The reasons for the lack of performance of the heuristics are: (i) the heuristics are based on link cost functions that find aggregation paths. Besides, these solutions rely on static rules to improve network lifetime via payload and header aggregation. For instance, in DAMiG and EESR, all traffic of a group must pass through a single node before going to an external group. This static rule is not efficient for a small number of groups. Therefore these heuristic solutions cannot achieve the optimal network lifetime; (ii) DAMiG does not achieve a higher lifetime performance because it always seeks to maximize the internal group traffic aggregation, which is not efficient in case of large groups; (iii) EESR does not apply header aggregation, which contributes to the poor lifetime performance.

TTAMA corrects the problem of DAMiG for the case of few groups (i.e. 1 and 2). When the number of groups is very small, the comparison shows that TTAMA achieves a performance that corresponds to more than 90% of the ILP. The main problem of TTAMA is related to the communication of external group traffic. As can be noticed, TTAMA does not maintain the same performance of ILP when there are many groups and the amount of external group traffic increases. It suggests that the static function used by TTAMA to find the best path to perform payload concatenation is not efficient.

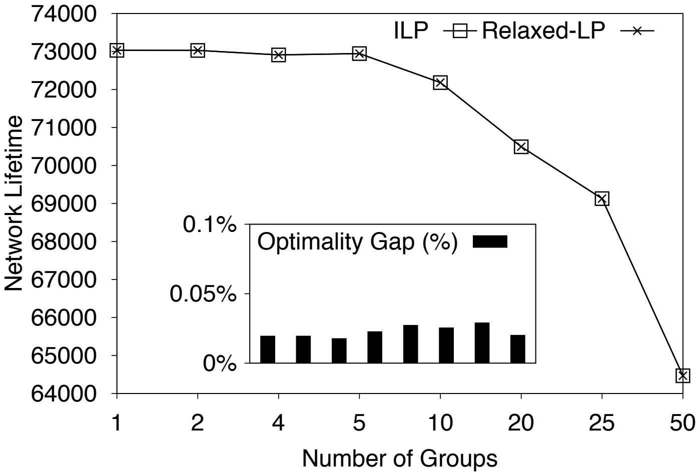


Figure 4.19: Optimality gap: difference between the relaxed model and ILP.

It is well-known that Integer Linear Programming problems belong to the class of Non-deterministic Polynomial (NP) problems. It means that unless P = NP, there is no polynomial-time algorithm for solving ILP problems. Although the complexity of the ILP problems, it is important to highlight that the proposed Integer Linear Program is computed once for the entire network lifetime, and it is used to compute the network lifetime upper bound. Thus, the ILP does not need to be periodically computed, which reduces considerably the amount of resources spent to find the maximum network lifetime.

A computer with 1,8 GHz Intel Core i5 and 4 GB of RAM was used to solve the proposed ILP model. Considering the case where there is a single group on a network of 100 nodes, the time to compute the optimal solution is on average 0.05 seconds. For the case where the network has 100 nodes and the number of groups is 10, the time to find the solution is 1.33 seconds.

Motived by the fact that the proposed ILP demands a considerable amount of computational resources, especially for large network sizes, it is important to present alternatives to solve the proposed optimization model in an affordable amount of time. One technique largely present in the literature to decrease the complexity of the ILP problems is to relax the integrality constraints. With this technique, instead of having constraints forcing the values of the variables to belong to the Integer set $\mathbb{Z}$, the values of the variables are allowed to belong to the Rational set $\mathbb{Q}$. Thus, the Integer Linear Program turns into a Linear Program (LP).

The relaxation of the ILP model raises the question of how large the gap between the optimal (i.e. ILP) and the relaxed-LP model is. In our case, the relaxed-LP model drops the integrality constraints and has $\lambda = 0$. Figure 4.19 shows the optimality gap between the two models. As can be seen, the optimality gap is not greater than 0.05 %. This low percentage certifies that the relaxed-LP is not distant from the optimal solution.

For the evaluation of larger network sizes, the relaxed-LP model was used. The network lifetime performance shown in Figure 4.20 considers networks with up to 4000 nodes, varying the size of the message headers. As can be seen, the header size has great influence on the network lifetime. This influence occurs because the header communication is the most demanding energy activity (see Figure 4.18), and longer headers reduce the payload size inside each message, which limits the payload concatenation and thus the network lifetime.

Considering network scenarios with multiple sinks, Figure 4.21 shows the network lifetime performance considering 1, 2, 3, 4, and 6 sinks, while the number of groups is 40, 20, 10, and 5. In this evaluation, the network has 400 nodes. As can be seen, the network lifetime is less influenced by the number of sinks when there are 5 groups. For all other scenarios, the network lifetime presents significant variation. The reason for it is that the nodes have an energy consumption constraint, and the increase on the number of sinks will not change the energy constraint. However, when the number of groups increases, a communication bottleneck is created near the sink. Thus, for the cases of 10, 20, and 40 groups, increasing the number of sinks mitigates the communication bottleneck, which improves the network lifetime performance.

After presenting the mathematical formulation of the upper bound network lifetime, the next section presents testbed experiments that executed the Two-Tier Aggregation on a real environment.

4.5 Testbed Implementation and Experiments

Testbeds are the proper environments to consider aspects that usually are hidden in simulations or models. In this direction, this section presents the implementation details and the experiments regarding the insertion of many payloads in a single message. The implemented solution presented in this section is called Messaging mULTIple Payloads LayEr (MULTIPLEx) and it is able to collect small size data blocks from low power devices in an efficient way, carrying these data blocks in the payload of a single message. The MULTIPLEx is a light-

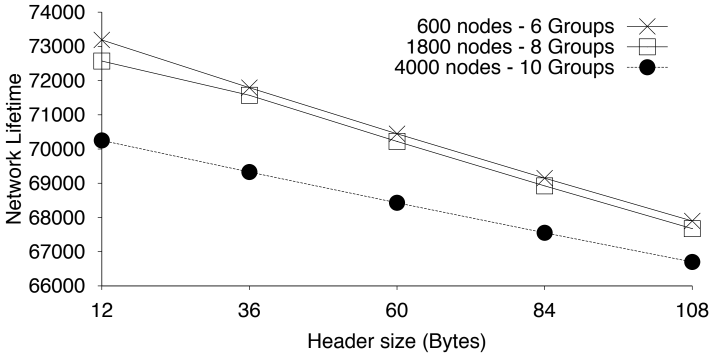


Figure 4.20: Network lifetime of large scenarios using different header sizes (Constant Network Size).

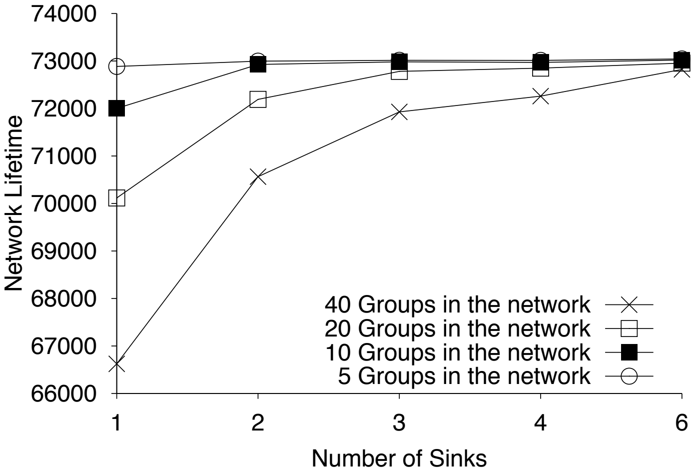


Figure 4.21: Network lifetime of Relaxed LP with Multiple Sinks (Constant Network Size - 400 Nodes).

weight layer designed to operate with the standard M2M protocol stack aiming to reduce the energy consumption of the energy constrained devices.

This section is structured as follows. Section 4.5.1 presents an overview of Messaging mULTIple Payloads LayEr (MULTIPLEx). Section 4.5.2 describes an experimental case study, including the software and hardware selection, the implementation details, and some preliminar results. Section 4.5.3 presents the evaluation and discusses the results.

4.5.1 Messaging Multiple Payloads Layer

MULTIPLEx allows the low power nodes to produce messages with multiple payloads when the node is producing its own messages and forwarding messages. The main idea of MULTIPLEx is to assemble messages with multiple payloads to exploit the fact that most of the periodic many-to-one traffic in M2M environments is composed by small size payloads.

As Figure 4.22 shows, MULTIPLEx is a layer located on top of the Internet Protocol (IP). During the forwarding process, the MULTIPLEx nodes store in a buffer the application messages received from the neighbors and also the self produced messages. The application messages are only stored if the node is not the final destination of the message. A timer is set periodically to verify if the MULTIPLEx criterion has been satisfied for issuing a multiple payload message. In case the criterion is positively verified, the payloads are extracted and information can be added to each individual payload, such as node id, datatype or timestap. Then, the extracted payloads are inserted as a bulked payload into a new application message. In case the criterion has not been satisfied, MULTIPLEx does not change any information in the messages. In both cases, the resulting messages are forwarded to the appropriate layer.

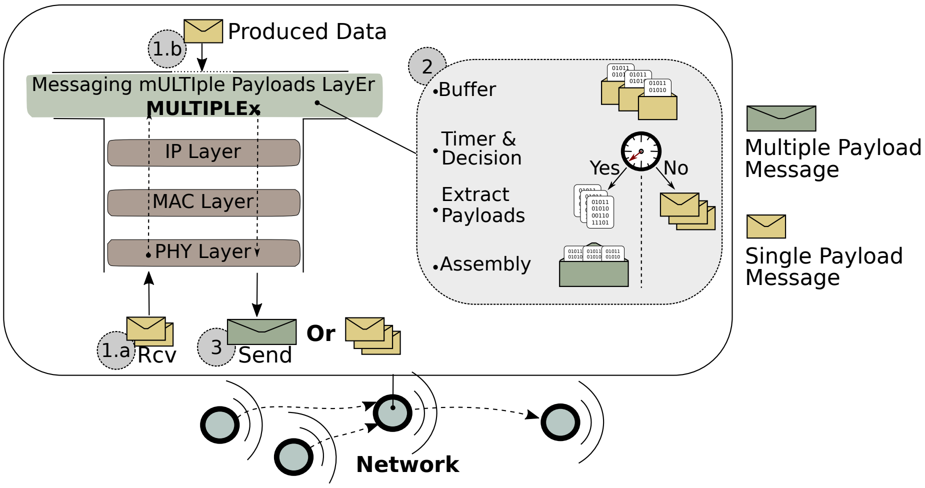


Figure 4.22: MULTIPLEx overview

MULTIPLEx can be designed with different criteria to decide if a new message will be created with multiple payloads or not. In this version of MULTIPLEx, it applies a simple, but effective criterion for this decision. The decision is based on a threshold, called δ, which is related to the number of stored messages. When the timer expires, MULTIPLEx counts the number of messages in the buffer. If this number is equal or greater than δ, then a new multiple payload message will be produced. Otherwise, all messages are preserved.

The decision based on the number of received messages is justified by the fact that the extraction and assembly operations demand a considerable amount of processing and energy resources. Therefore, MULTIPLEx is designed to perform these operations when the energy savings due to the message header suppression payoff the costs.

The next section shows detailed information about the implementation of MULTIPLEx in real devices.

4.5.2 Experimental Prototyping

This section presents the details of the prototyping implementation where MULTIPLEx is embedded on real devices. To begin, the next section shows the details of the software, hardware platform and implementation.

4.5.2.1 Implementation of MULTIPLEx

MULTIPLEx can be applied to different M2M application protocols, for instance CoAP [Shelby et al., 2014] and Message Queuing Telemetry Transport (MQTT) [Banks and Gupta, 2015]. For this case study, CoAP has been selected as the application protocol because it supports natively many-to-one traffic and has the "*observe*" option that enables a client to receive data periodically.

Contiki has been selected as the operating system. Some reasons that motivated this choice are: (i) Contiki provides stable open-source version of the IEEE standard protocols; (ii) it has a large community of developers; and, (iii) it is designed for several low-energy hardware platforms.

Regarding the hardware, the Contiki nodes run on the ATmega256RFR2-XPRO board[3]. It has a 16 MHz Micro-Controller integrated with a 2.4Ghz transceiver, 256 KBytes of flash, and one temperature sensor.

MULTIPLEx is implemented on Contiki as a light-weight layer, located between the Internet Protocol (IP) and the User Datagram Protocol (UDP) layers. MULTIPLEx was implemented between these layers in order to verify the destination IP address of every message received from the neighbors. This verification is necessary because payloads are extracted for the creation of new CoAP messages if the message has not reached the final destination. For the case of a CoAP message arriving from the UDP layer, MULTIPLEx assumes that this message has been produced by the node itself, so it does not extract the payload from this message.

Algorithm 4.4 shows the main logic executed by MULTIPLEx when a new packet arrives from the IP layer and also presents the algorithm executed when the timer triggers the decision of creating a new multiple payload message.

Algorithm 4.4. MULTIPLEx algorithm

```
 1: Initialize: threshold
 2: Start
 3:     function NEW_PACKET_ARRIVES(msg)
 4:         typeCode ← parseCode(msg)
 5:         address ← parseAddrs(msg)
 6:         if isCoAP(typeCode) and notMyAdd(address) then
 7:             buffer(msg)
 8:             if timer_not_set() then
 9:                 set_timer()
10:             end if
11:         end if
12:     end function
13:
14:     function TIMEOUT_CALLBACK( )
15:         msg_num ← count_msg(buffer)
16:         if isGreater(msg_num, threshold) then
17:             payloads_data ← extractPayloads(buffer)
18:             new_coap ← assembly_new_coap(payload_data)
19:             forward_next_layer(new_coap)
20:         else
21:             forward_next_layer(buffer)
22:         end if
23:     end function
24: End
```

In Algorithm 4.4, from line 3 to 12, MULTIPLEx implements a simple buffer for messages arriving from the IP layer and sets a timer. In lines 14 to 23,

---

[3] http://www.atmel.com/tools/atmega256rfr2-xpro.aspx

MULTIPLEx decides if it will create a new CoAP message having multiple payloads. Depending on the number of stored packets on the buffer, the creation of a multiple payload message is cost-efficient or not. For this reason, it is necessary to set the threshold δ with an adequate value. With this purpose, Section 4.5.2.3 presents a study to determine a value to δ.

The next section shows the experiments conducted to calibrate and validate a software tool used to estimate the energy consumption of the nodes.

4.5.2.2 Calibration and Validation the Contiki's Energy Tool

Contiki contains an Energy Consumption Tool, called Energest, that is able to estimate how much energy a node has consumed. The use of Energest is motivated by the fact that hardware-based energy measurement is time consuming and requires many dedicated boards and equipment to measure the energy consumption of few nodes.

Energest uses a Contiki service that maintains a table with the total time the CPU and transceiver have been active. This service produces time stamps when the component is turned on and off. Having these time stamps, it is possible to estimate the energy consumption of the node, as presented in Equation 4.17.

$$\frac{E}{V} = I_m t_m + I_t t_t + I_r t_r \tag{4.17}$$

In Equation 4.17, t_m, t_t, and t_r are the time the micro-controller is in the following states: active, transmitting, and receiving, respectively. Besides, the set of constants $\{I_m,\ I_t,\ I_r\}$ represents the electrical current necessary to run each of these states. To achieve an accurate estimation of the total energy consumption via Energest it is necessary to calibrate the values of the set $\{I_m,\ I_l,\ I_t,\ I_r\}$. This calibration was performed using a two step methodology. First, real measurements were performed on the Contiki devices to find the electrical current in each state. Second, the total energy consumption estimation provided by Energest was compared to the real measurements.

The MSP432 P401R[4] board was used to measure the real current energy consumption of the ATmega256RFR2-XPRO boards. The P401R board can measure the amount of energy consumed and the electrical current of a target board with a resolution of 2kHz.

The energy states have been measured in 10 different ATmega256RFR2-XPRO boards and executed twice for each board. Each test lasted 10 minutes and the first two minutes were considered as a warm-up period. Table 4.8 shows the obtained values of the current for each state.

CPU measurements were performed running Contiki without entering in low power mode, since this version of Contiki is not stable when ATmega256RFR2

---

[4] http://www.ti.com/lit/ug/slau597c/slau597c.pdf

Table 4.8: Current Measurement of the States

State	**CPU**	**Transmission**	**Reception**
Electrical Current	9.32 mA*	21.44 mA	15.49 mA

*This measurement includes idle and active.

is in low power mode. The result for CPU is the average electrical current for the whole period of the experiment, in which the transceiver is turned off and the CPU can be active or idle.

Table 4.9 shows the real and the estimated energy consumption reported by the P401R board and the Energest, respectively. For these tests, 10 boards were subject to three separated executions of 1 hour of operation, which comprises the transmission to a border router of one CoAP message every 30 seconds. The error column in Table 4.9 shows in percentage how much the estimated energy is different from the real measurements.

Table 4.9: Total Energy Consumption

Board	**Real**	**Energest**	**(Error %)**
1	112707.3	115272.3	2.28
2*	125105.1	127952,3	n/a
3	108492.0	114006.3	5.09
4	108710.0	114345.0	5.19
5	111113.0	114455.0	3.01
6	109282.3	113216.6	3.60
7	113827.7	113929.6	0.20
8	111921.7	114646.3	2.44
9	109843.7	114412.0	4.16
10	111296.3	114086.0	2.50
Avg	110799.3	114263.2	3.16

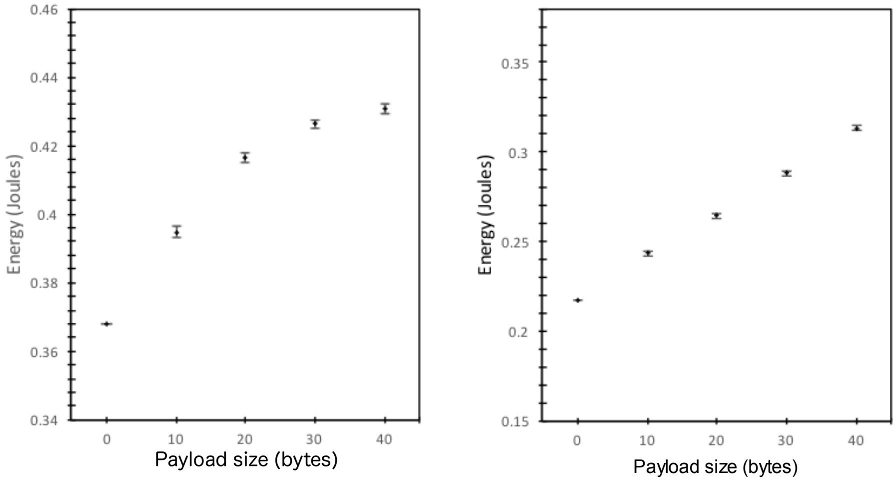
*Outlier.

Hurni et al [Hurni et al., 2011] present measurements related to the calibration of the Contiki's energy consumption tool. The obtained average error value of 3.16% is compatible with the error interval presented by Hurni et al.

The next section presents some prelimilar experiments performed on a reduced set of nodes.

4.5.2.3 Preliminar Tests and Study of Threshold δ

The first preliminar set of tests aims to study the energy consumption of a node when it receives or transmits messages with different payload sizes. Figures 4.23a and 4.23b show the amount of energy consumed for reception and transmission, respectively. These experiments were conducted 30 times and the average of data samples was obtained considering a confidence interval of 95 %. As can be observed, the consumption due to transmission tends more to a linear curve than the reception. In both cases, it is possible to see the impact the payload size has on the consumed energy. Besides, these figures show the amount of energy spent only to transmit or receive the header. These amounts are saved by MULTIPLEx every time the nodes use a single header to communicate payloads.



(a) Reception.

(b) Transmission.

Figure 4.23: Energy Consumption.

Another objective is to study the threshold δ, which is the minimum number of payloads on a message able to be cost-efficient in terms of energy consumption. The study of the threshold δ was conducted using four nodes: 1 node running MULTIPLEx, 2 nodes that inject CoAP traffic, and 1 receiver node. For comparison purposes, the standard CoAP has been evaluated in the same scenario. Standard CoAP produces messages containing only a single payload and forwards all the received messages from the 2 injector nodes. Figure 4.24 shows the obtained results, considering a communication round of 120 seconds.

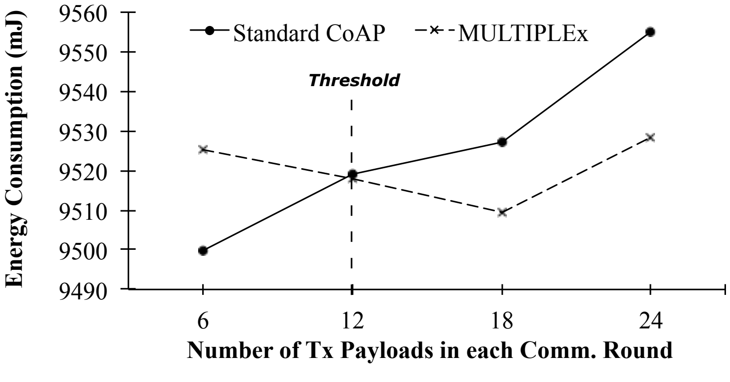


Figure 4.24: Threshold δ

As can be seen in Figure 4.24, MULTIPLEx has a lower energy consumption if the number of payloads in the message is equal or greater than 12. Otherwise, it is more efficient to use the standard CoAP, in which the messages have a single payload. The main reason MULTIPLEx does not payoff for less than 12 payloads is related to the cost of extracting and producing a new message.

The next section describes and discusses the results obtained from the experimental evaluation.

4.5.3 Testbed Evaluation and Results

To evaluate the energy consumption of MULTIPLEx, a testbed composed of 8 Contiki nodes was used. As depicted in Figure 4.25, the 8 Contiki nodes were deployed to create a balanced tree topology in an indoor environment.

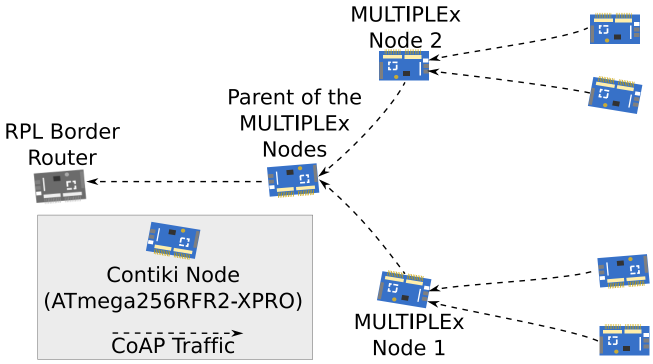


Figure 4.25: Testbed Topology

The experiments were conducted to evaluate MULTIPLEx and the Standard CoAP solution under the same settings. The following set of protocols was used: CoAP, UDP, 6LowPAN, RPL, ContikiMac, and IEEE 802.15.4. All obtained results were computed based on 15 independent experiments. Each experiment lasted for 10 minutes, and the first 4 minutes were defined as the warm-up internal. The measurements were obtained by the Contiki's Energy Consumption Tool (see Section 4.5.2.2).

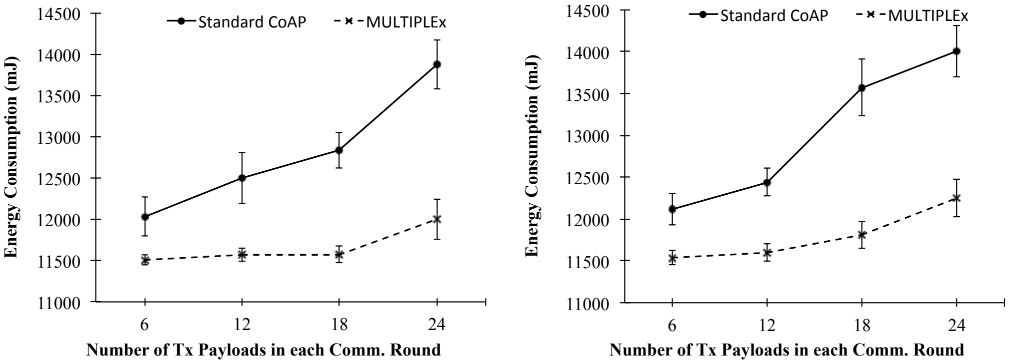
The final destination of all injected CoAP messages is an external computer, which is connected to the border router and runs a traffic capture software to collect the received messages. Table 4.10 presents additional seetings used in the experimental evaluation.

Table 4.10: Settings of the Real Experiments

Setting	**Value**
MULTIPLEx Threshold	12 Payloads
Application Protocol	CoAP
Single Payload Size	2 Bytes
MAC Protocol	ContikiMac
ContikiMac Channel Check Freq.	8hz
Wireless Technology	IEEE 802.15.4
Communication Round Interval	60 sec

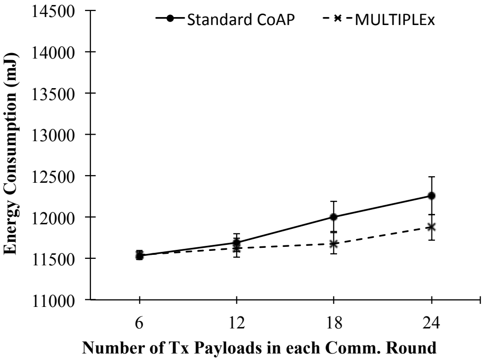
Among the 8 nodes, 1 node is the border router, 1 node acts as parent for the rest of the network, 2 nodes execute the MULTIPLEx code, and 4 nodes inject CoAP messages in the network. The parent node is set to forward the CoAP messages received.

Figure 4.26 presents the average total energy consumption and the standard deviation for different numbers of CoAP payloads. In Figures 4.26a and 4.26b, the results show the energy consumption of the two MULTIPLEx nodes. Regarding these results and for all measured cases, MULTIPLEx has the lowest energy consumption. The improvement is 14.85% for the case of 24 payloads in a single message.



(a) MULTIPLEx Node 1

(b) MULTIPLEx Node 2



(c) Parent Node

Figure 4.26: Energy Consumption

In Figure 4.26c, the obtained energy consumption corresponds to the node that acts as parent for the MULTIPLEx nodes. In this set of results, MULTPLEx is more cost-effective in the case of 18 and 24 payloads. This figure shows that MULTIPLEx also improves energy consumption of the parent node, since this node forwards less CoAP headers.

Table 4.11 shows the average energy consumed in each state of the MULTIPLEx nodes. In these results it is possible to measure how MULTIPLEx changes the energy spent on Transmission, Reception, General CPU operation, and also it presents the CPU demanded by MULTIPLEx.

The results in Table 4.11 show that MULTIPLEx compared to the standard CoAP solution increases the energy spent by the CPU. However, the MULTIPLEx CPU consumption is cost-effective, since the saved energy on Transmission and Reception is greater than the energy consumed to execute MULTPLEx. Another advantage of MULTIPLEx is that it preserves the content of the payloads, which means that the data accuracy is not reduced.

Table 4.11: Energy Consumption

	Tx Payloads	**General CPU**	**Tx**	**Rx**	**CPU for MULTIPLEx**
MULTIPLEx	6	95.50%	1.09%	3.36%	0.03%
	12	94.80%	0.97%	4.14%	0.07%
	18	93.30%	1.16%	5.41%	0.11%
	24	89.40%	1.41%	9.03%	0.14%
Standard	6	91.32%	2.97%	5.70%	n/a
	12	88.20%	6.06%	5.73%	n/a
	18	83.27%	6.21%	10.51%	n/a
	24	78.90%	9.46%	11.63%	n/a

4.6 Summary

This chapter proposed mechanisms to address the following main open issues in M2M communication: (i) Energy efficiency in limited battery-power networks, (ii) heterogeneous group communications. It presented, in Section 4.2, a flexible and general approach to reduce overhead and eliminate data redundancy that can be applied in M2M group communications, called Two Tier Aggregation (TTA).

Besides, Section 4.3 presented two heuristics designed to apply the proposed TTA over the network traffic, focusing to compute how the messages will flow on the network to achieve better network lifetime performance via TTA.

Section 4.4 introduced a mathematical model, formulated as an Integer Linear Program, able to find the optimal network flow solution for TTA that maximizes the network lifetime. In addition, this section presented a performance comparison with the objective to determine how far are the heuristics from the optimal or near-optimal solutions.

Section 4.5 showed the prototyping and the testbed experiments. These set of experiments were conducted to measure the real gain of the proposed TTA. The obtained result highlighted that the aggregation cost is not negligible, but it can be considered as a criteria to decide if the aggregation should occur or not.

The next chapter shows the proposed solutions for energy efficient communication in replenishable energy environments.

Chapter 5

Dynamic Data Aggregation in Replenishable Energy Environments

> If I have ever made any valuable discoveries, it has been owing more to patient attention, than to any other talent.
>
> *(Isaac Newton)*

Contents

To use energy harvesting sources to replenish the energy reserves of Machine-to-Machine (M2M) devices opens relevant research challenges. One of the these challenges is to consume energy resources in a pace that does not deplete or overflow the node's reserves. Another challenge is to reduce the communication overhead, since it contributes for the network congestion and waste of energy resources. In addition, the machine-to-machine standard protocol stack must be adapted and extended for energy harvesting scenarios, promoting the convergence of protocols in this field. Motivated by the these challenges, this chapter presents two proposed mechanisms designed for replenishable environments.

5.1 Introduction

Energy harvesting enables the network nodes to renew their energy reserves using solar, wind, or motion power. Due to the capacity of renewing energy, energy harvesting has demanded new insights on how to communicate efficiently. One important requirement in energy harvesting networks is to regulate the energy consumption according to the level of energy stored in the batteries, avoiding the over and underuse of the energy resources [Gunduz et al., 2014]. Several proposals are designed to this regard. For instance Gregori and Payaró [Gregori and Payaró, 2012] present some transmission strategies that compute the amount of data to be communicated according to the energy availability.

Data aggregation is an approach, usually coupled with routing, that has a great potential to regulate the energy consumption. This is because data aggregation allows the nodes to control the amount of spatio-temporal redundancies flowing over the network paths. Using data aggregation, the sink node can receive the aggregated data that reveals the overall monitoring status (e.g. average temperature), instead of receiving the individual data readings from each node. Thus, as the nodes aggregate data, the traffic flow can be reduced over the network.

When the nodes do not have energy harvesting capabilities, it is necessary to aggregate the data as much as possible, keeping the maximum aggregation level during the whole network lifetime. However, in energy harvesting networks, there are moments when it is possible to spend more energy, requiring the nodes to regulate dynamically their aggregation level. For instance, after a time interval of high solar irradiation, the nodes can increase their energy consumption, allowing the nodes to store future harvested energy.

At first, the idea of dynamically regulating the level of data aggregation according to the energy level can be seen as undesired for some users and companies, because the amount of collected data will depend on the energy harvested from the ambient. However, most of the classic data aggregation solutions offer only one data aggregation level, which results in minimum data accuracy. So, there is a gain when dynamic data aggregation is executed, since in the best case scenario the highest data accuracy level is achieved. Only in the worst case scenario the minimum data accuracy level is achieved.

Regarding the argument that the harvested energy from the ambient is uncontrollable, a straightforward solution to reduce the influence of natural conditions on dynamic data aggregation is able to introduce upper and bottom thresholds for the data aggregation level. Using thresholds, the variation of the data aggregation level is limited and can be adapted to the necessity of the users.

Despite the aforementioned aspects of energy harvesting and data aggregation, the current solutions for energy harvesting networks do not adapt the aggregation level in accordance to the amount of energy harvested in the batteries. To fill this gap, this chapter presents:

- A distributed data aggregation solution, named Data Aggregation for energy harVesting NETworks (DAV-NET), that dynamically controls the level of data aggregation and adapts the energy consumption according to the level of energy in the batteries.
- A Routing and Aggregation for Minimum Energy (RAME) energy-aware solution that combines routing and data aggregation mechanisms to control the data traffic on the paths considering lowest energy reserve on the path. In addition, this solution is implemented on the standard M2M protocol stack and applies a scheme that successfully reduces the overhead of the communication.

The remaining of this chapter is structured as follows: Section 5.2 presents the DAV-NET approach and shows the obtained results. Section 5.3 describes the RAME approach and presents the implementation and evaluation.

5.2 Data Aggregation for Energy Harvesting Networks

This section outlines and evaluates the Data Aggregation for energy harVesting NETworks (DAV-NET). Unlike traditional data aggregation solutions, DAV-NET is designed to dynamically adapt the data aggregation level, seeking the neutral operation of the nodes.

To begin, the next section presents an overview about DAV-NET.

5.2.1 Overview

The aim of the proposed solution, named Data Aggregation for energy harVesting NETworks (DAV-NET), is to route the data over the paths promoting data aggregation to use in an efficient manner the harvested energy stored in the batteries.

Figure 5.1 shows a solar scenario, where a network node has a buffer for input data, which stores own produced data and received messages from neighbors. This particular node computes the aggregation level according to the energy level of the battery, which will depend on the harvested and consumed energy. The main idea is to regulate the amount of traffic flowing through the node using the aggregation level as parameter. In other words, the node uses the aggregation level to determine the number of payloads in the buffer that should be aggregated.

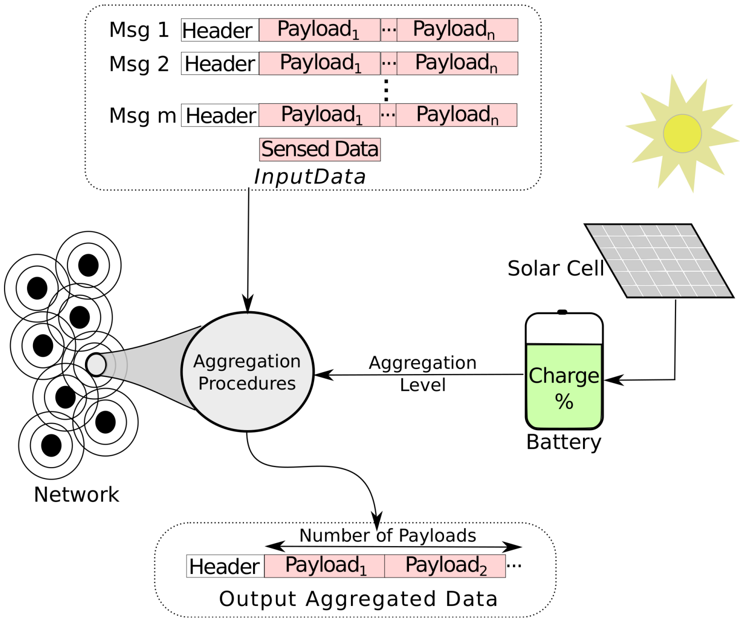


Figure 5.1: Data Aggregation in DAV-NET.

In general, DAV-NET enables the network to decrease the number of aggregated payloads when the energy harvested from the ambient increases the energy level in the batteries. In case the node has few energy resources, the number of aggregated payloads will increase. This traffic regulation allows the network to make efficient use of the energy resources.

The next section further details the design of DAV-NET.

5.2.2 Energy Harvesting with Data Aggregation

This section presents the main steps and procedures executed by DAV-NET to dynamically aggregate data in energy harvesting environments. As presented in the following section, first, it computes an aggregation tree, allowing the traffic to flow from the nodes to the sink. Second, it dynamically finds an aggregation level that regulates the aggregation procedures executed on the network traffic.

5.2.2.1 Aggregation Tree

DAV-NET computes an aggregation tree in a distributed fashion. To achieve that, each node maintains the following information about its neighbors: the *energy level* and the *hop gradient*. The former reveals the battery state-of-charge, and the latter is the number of hops necessary to reach the sink node. Data gathering is executed in communication rounds. In each communication round, all nodes communicate their data readings to the sink node.

As can be seen in Algorithm 5.1, during the Beacon Interval, which is executed periodically, every node broadcasts beacon messages containing the energy level, and the hop gradient. Through the beacon messages the nodes can update the neighbors' information. Even in stationary topologies, the periodic broadcasting of hop gradient is necessary due to network dynamics (e.g. node and link failures and energy depletion).

Algorithm 5.1. Data Aggregation Tree

```
Initialize:
ownHop, ELevel, txtimer, rxtimer;
Start
    while isBeaconInterval do
        TXtimer ← findtxTimer();
        RXtimer ← findrxTimer();
        if isRXtime(RXtimer) == True then
            bMSG ← listenNeighbors();
            update_ownHop(bMSG);
            update_ngbHop(bMSG);
            update_ngbElevel(bMSG);
        end if
        if isTXtime(TXtimer) == True then
            ownHop ← getOwnHop();
            Elevel ← getEnergyLevel();
            bMSG ← createBeacon(Elevel, ownHop);
            broadcast(bMSG);
        end if
    end while
    selectParent();
    computeAggLevel;
End
function selectParent
parentID ← 0;
    do
        parentID ← nextParentMaxE(parentID);
        nHop ← getHopGradient(parentID);
    while nHop > ownHopG
end function
```

At the end of the Beacon Interval, each node has the information revealing its own and the neighbors' energy level and hop gradient. Having this information, each node begins the process of parent selection, which is a straightforward procedure. First, the selector node chooses as parent candidate the neighbor with highest energy level. Second, the parent candidate becomes parent if the hop gradient of the parent candidate is less than the hop gradient of the selector node. If the parent candidate violates the hop gradient test, the selector node finds the neighbor that has the second highest energy level, repeating this procedure untill it finds a node that passes the hop gradient test. This procedure is based on the Routing Protocol for Low-Power and Lossy Networks [Winter et al., 2012], and avoids loops in stationary topologies.

As the nodes select as parent the neighbor with highest energy level, the nodes with low energy level tend to be children in the tree, consuming less energy than the parent nodes. The next section gives the details about the procedures executed by DAV-NET to aggregate the network traffic.

The next section shows how the aggregation level is computed by each node.

5.2.2.2 Dynamic Aggregation Level

To find the Aggregation Level of a node i, DAV-NET uses Equation 5.1.

$$AggLevel_i = 1 - \left(\frac{E_i}{E_{battery} * \alpha}\right) \tag{5.1}$$

E_i is the residual energy of node i and $E_{battery}$ denotes the maximum capacity of the energy reserve. The α parameter is defined to avoid the battery to reach its maximum limit, and $0 < \alpha < 1$. In general, α is determined according to the battery type, and it is widely defined in the charging control circuits to avoid overcharge. For instance, Nickel-Cadmium (NiCd) and Nickel-Metal Hydride (NiMH) batteries are considered fully charged if the residual energy is greater than 80% of the total capacity [Hussein and Batarseh, 2011].

The next step is to compute the amount of data that should be aggregated and sent. For this purpose, each parent node $i \in N$ maintains a buffer, denoted as $InputData_i$, that stores the messages received from its children and its own sensed data. Then, Equation 5.2 is used by a node $i \in N$ to determine the number of payloads composing the output aggregated message, in a given communication round k. In this equation, $inPayloads(InputData_i)$ returns the total number of payloads extracted from the buffer (including the sensed data).

$$outPayloads_i^k = inPayloads(InputData_i) * (1 - AggLevel_i) \tag{5.2}$$

As $outPayload_i^k$ is a real number, DAV-NET rounds the number of output payloads if it is less than 1.

DAV-NET is able to compute the aggregation level of a node according to harvested energy stored in the battery. Thus, in moments of plenty of energy, the data aggregation is reduced, resulting in more network traffic and energy consumption. On the other hand, in the periods of no or low harvested energy, the data aggregation increases, promoting a reduction on the network traffic and consequently decreasing the energy consumption. Due to the concatenation of payloads, DAV-NET enables the network to regulate the energy consumption without increasing the energy consumption due to message headers overhead.

The next section gives more information about the aggregation procedures executed on the traffic.

5.2.2.3 Data Aggregation Procedures

It is considered that each message in $InputData_i$ has two parts, the header and the application layer payload, or simply payload. Besides, the size of a single payload is small enough for the creation of an aggregated message containing multiple payloads.

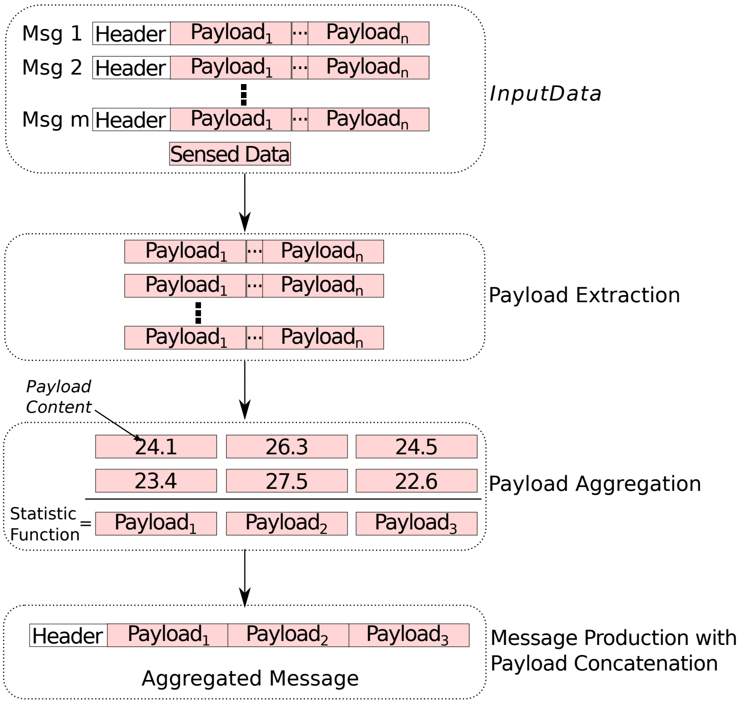


Figure 5.2: Aggregation Procedures.

As depicted in Figure 5.2, a parent node i aggregates the messages in $InputData_i$ executing the following Aggregation Procedures:

- **Payload Extraction:** This Aggregation Procedure extracts the payloads from all the messages stored in $InputData_i$, discarding the headers.
- **Payload Aggregation:** Uses statistics functions to process the data contained in the payloads. Several aggregation functions (e.g. Sum, Maximum, Minimum, and Average) can be used in this aggregation procedure. We assume that all nodes use the same aggregation function for Payload Aggregation, and this function is defined by the Application User.
- **Payload Concatenation:** It creates a single message containing many payloads.

In every communication round, every parent node $i \in N$ executes the Payload Extraction on all data stored in the $InputData_i$ buffer. The Payload Extraction eliminates the headers, it does not change the amount of data carried on the payloads, as its function is to deliver to the Payload Aggregation a set of Payloads.

Unlike the procedures executed on Payload Extraction, the Payload Aggregation procedure processes the data on the payloads, enabling the production of new payloads. The Payload Aggregation is able to reduce a given subset of payloads into a single payload.

After the Payload Extraction and Aggregation, the last procedure is the Message Production with Payload Concatenation. This procedure allows the DAV-NET to reconstruct the aggregated payloads containing the new payloads generated by the Payload Aggregation.

Using these Aggregation Procedures, DAV-NET can dynamically regulate the amount of network traffic flowing over the network according to the energy level.

After showing the details about the proposed solution, the next section presents the performance evaluation and the obtained results.

5.2.3 Performance Evaluation

This section presents the performance evaluation of the proposed solution, showing the obtained results when DAV-NET regulates the network energy consumption using the proposed aggregation tree, the aggregation procedures and the aggregation level.

5.2.3.1 Objectives

The aim of this evaluation is to show the impact that DAV-NET has on the network energy consumption in a daily basis. The expected result is to achieve different profiles of consumption depending on the aggregation level applied on the network traffic.

5.2.3.2 Evaluation Settings

For the performance evaluation, we use a random network topology, where a sink is located in the center and 100 nodes are uniformly distributed in an area of 180x180m. Besides, the assigned value for $E_{battery}$ corresponds to two typical AA batteries, which means 30700J. This energy can be computed knowing that a single AA battery has 2.85 Ah, which corresponds approximately to 15350J if the voltage is fixed on 1.3V. The values of the other parameters used for the performance evaluation are shown in Table 5.1.

Regarding the energy consumption model, the nodes have three communication states, namely transmission (Tx), reception (Rx), and sleep. It is well-known that the transition between the communication states also consumes energy. Thus, s_{trans} denotes a single transition from Rx or Tx to sleep, or vice-versa, and according to Ma et. al [Ma et al., 2009] we consider $s_{trans} = 21\mu J$.

Table 5.1: Performance evaluation parameters and values.

Parameter	**Value**
Header Size	12 Bytes (CoAP) + 36 bytes
Default Payload Size	4 Bytes
Maximum Transmission Unit	127 Bytes
Communication Round Periodicity	1 second
Number of Network Nodes	100
Beacon Interval Periodicity	10 Communication rounds
Battery Capacity	30700J (2 AA Batteries)
α	0.75
Maximum Wireless Range	30 m

Owing to the energy costs incurred in these transitions, the nodes change once from sleep mode to Tx or Rx mode, and afterwards, change once again from Tx to sleep mode, spending $2 * s_{trans}$ in a single communication round. In addition, since it can be assumed that the devices are able to maintain perfect clock synchronization, the energy spent on overhearing and idle states is not considered.

Denoting wrx_i^k as the total number of bits received and wtx_i^k as the number of bits of the single message transmitted by i in the communication round k. Each node $i \in N$ knows its parent $j \in N$, and the Euclidean distance $d_{i,j}$ between i and j. Finally, the energy consumed by a node i due to communication activities during a single communication round k is given as:

$$Ecomm_{i,k} = e_{amp} * d_{i,j}^2 * wtx_i^k + e_{elec} * wrx_i^k + 2 * s_{trans} \tag{5.3}$$

In Equation 5.3, the constants denoted by e_{amp} and e_{elec} represent the energy necessary to amplify and to run the electronic circuit for a single bit, respectively. The first two terms of Equation 5.3 represent the energy consumption related to the data transmission and reception, and the third term measures the energy consumption caused by the transitions between different communication states.

Regarding the amount of energy harvested from the environment, it is considered that in a single day the solar irradiance is able to charge the battery during 4 hours. As Zhang et al. [Zhang et al., 2011a], we assume that the harvested energy is uniformly distributed over the network, using a Gaussian function to vary from 0 to 100mWh the harvested energy in each communication round. All the evaluated results were obtained based on 30 samples, considering a simulation time of 6 days.

As the proposed solution varies the data aggregation according to the battery level, DAV-NET is evaluated in three different battery state-of-charge, detailed as follows:

- The Low State-of-Charge (LSC) corresponds to 5000J of energy, which is

17% of $E_{battery}$;

- The Intermediate State-of-Charge (ISC) has 10000J of energy, which is 35 % of $E_{battery}$;
- The High State-of-Charge (HSC) has 15000J of energy, corresponding to 50 % of E_{batery};

5.2.3.3 Results

Figure 5.3 shows the residual energy stored in the batteries over the simulated time. As expected, in all the cases, the nodes renewed their energy reserves. On average the nodes harvested 6000J (i.e. approximately 20% of $E_{battery}$) from the ambient during the simulated 6 days. At the end of the 6th day, HSC, ISC, and LSC achieve approximately 21000J, 14000J, and 9000J, respectively. At this moment, HSC almost approaches the maximum permissible residual energy, which is around 23000J (i.e. app. 75% of $E_{battery}$).

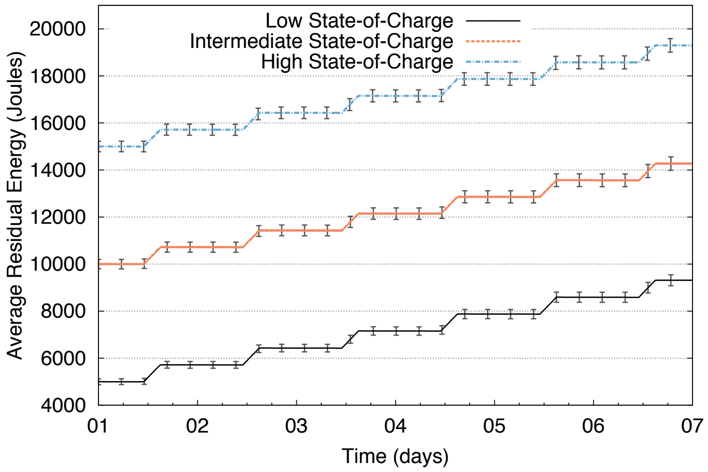


Figure 5.3: Average Residual Energy.

Figure 5.4 shows the total number of payloads that were aggregated by all nodes during each communication round. As can be observed, due to the dynamic data aggregation control, the number of aggregated payloads tends to follow the inverse residual trend. In general, the number of aggregated payloads decreases over time, since the battery level increases. While the network aggregates on average almost 162 payloads for LSC, it aggregates on average 35 payloads when the nodes are in HSC.

Figure 5.5 presents the average total number of payload transmissions executed by all nodes during each communication round. Each node executes at least one payload transmission in each communication round, so the minimum number of transmitted payloads for each round is 100 (i.e. one transmission per node). The minimum number of payload transmissions is achieved when the data aggregation is maximum, while the maximum number of payload transmissions is achieved when the aggregation level is minimum.

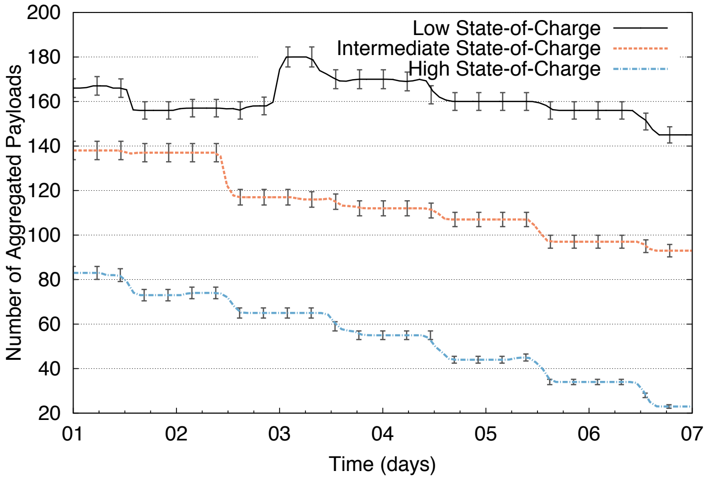


Figure 5.4: Sum of aggregated payloads.

Knowing that DAV-NET increases the number of payload transmissions according to the residual energy, when the nodes have the lowest residual energy in LSC, the whole network transmits on average 102 payloads. Without violating the limits of IEEE 802.15.4, the sum of all payload transmissions of HSC in each communication round is 450 payloads at the 6th day. It is important to notice that the 450 payloads do not correspond to the number of payloads received by the sink node, but it corresponds to the sum of all transmitted payloads of all nodes in a given communication round.

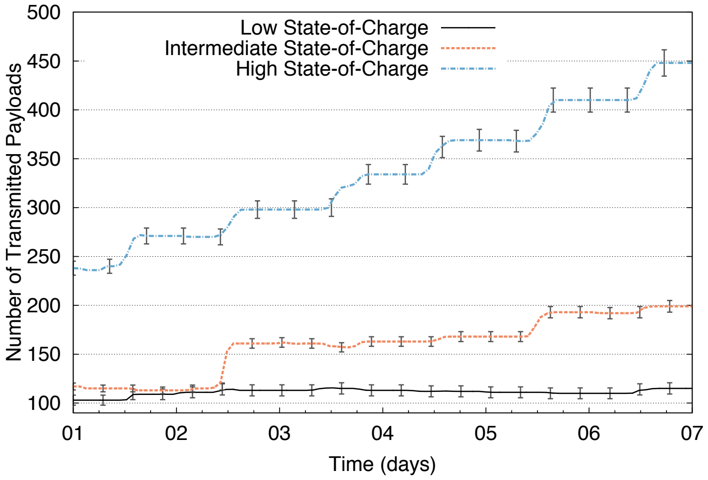


Figure 5.5: Transmitted payloads.

Figure 5.6 shows the average amount of energy that a node consumes during a single day. It is clear that the DAV-NET execution impacts the energy consumption. These results show that DAV-NET can impact the daily energy consumption of a node in approximately 1.5 Joules. This value represents a valuable amount of energy for the long term operation of the network (i.e. several months). It also shows that data aggregation is a suitable mechanism for regulating the energy consumption in energy harvesting networks.

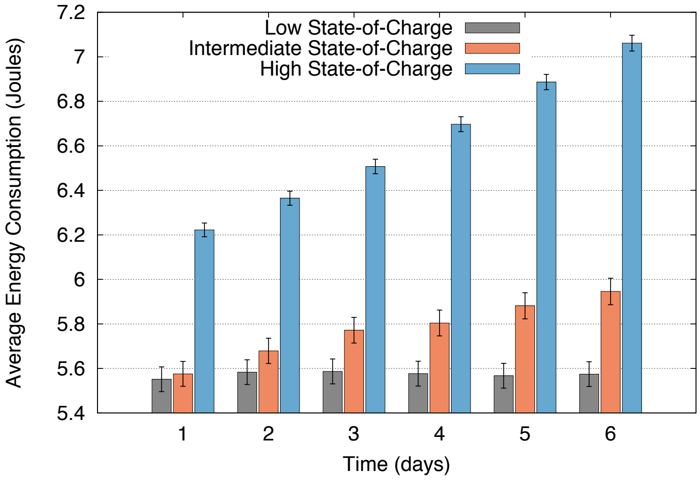


Figure 5.6: Average Cumulative Energy Consumption.

In addition, Figure 5.6 depicts that when DAV-NET is in Low-State-of-Charge the energy consumption tends to be in 5.5 Joules per day. The reason for these results is in the fact that the number of transmitted payloads is kept around in 100, which means that the network traffic is regulated at a constant level.

An important aspect observed in the DAV-NET performance is that it increases or decreases the energy consumption through the number of payloads. It means that rather than merely increasing the number of messages on the network, the proposed solution regulates the communication of valuable information (i.e. payloads). As consequence, it does not increase the communication of headers.

The next section presents another solution also designed for energy harvesting environments.

5.3 Routing and Aggregation for Minimum Energy

The results of DAV-NET show that a dynamic data aggregation solution can be used to achieve different energy consumption profiles, which is useful in energy harvesting environments. The proposed solution in this section, Routing and Aggregation for Minimum Energy (RAME), improves some drawbacks of DAV-NET and also uses some of its insights. For instance, instead of using the percentage of the residual battery to determine the aggregation level, RAME computes the aggregation level based on the minimum energy reserve on the path, which is the energy bottleneck of the route. So, while DAV-NET is focused on regulating individually for each node the data aggregation level, RAME takes into consideration the amount of energy in the route to decide the data aggregation level and to select paths.

This section is structured as follows: Section 5.3.1 shows an overview about the proposed solution. Section 5.3.2 introduces the main components of RAME. Section 5.3.3 outlines the aspects related to the implementation. Section 5.3.4 presents the conducted evaluation and the obtained results.

5.3.1 Overview

Neutral Operation or Energy Self-Sufficient are terms widely used to define energy harvesting efficiency. It means that a node consumes the exact amount of harvested energy [Ozel et al., 2015]. When a node is in neutral operation, there is no overuse or underuse of the energy on the battery. However, to achieve neutral operation of a multi-hop network is a very complex problem, since the nodes do not have full knowledge of the network and not all nodes have the same harvesting pattern, energy consumption, and battery capacity [Riker et al., 2015b].

A simplification of the Neutral Operation problem for multi-hop energy harvesting networks can be achieved if the operation of the whole network is determined by what can be supported by the node with the lowest amount of energy. This simplification, called Neutral Operation of the Minimum Energy Node (NOMEN), guarantees that all network nodes will cooperate to turn the operation of the minimum energy node into a neutral operation.

The solution proposed for the NOMEN problem is Routing and Aggregation for Minimum Energy (RAME). RAME is a joint battery-aware solution that encompasses routing and data aggregation. Regarding routing, it selects the path with the maximum lowest energy reserve, which is a max-min strategy. Relatively to data aggregation, RAME applies procedures that enable a message to be assembled with multiple payloads, and in case there is a need to reduce traffic, it aggregates the traffic load according to the minimum energy reserve on the path [Riker et al., 2016a]. The main idea behind RAME is to select paths where the minimum energy node has the highest possible energy reserve and to use data aggregation to regulate the traffic load in each path.

RAME contains three components, as depicted in Figure 5.7 and described as follows:

1. Battery Model: A component able to estimate the remaining energy charge in the battery. It periodically increments the amount of energy harvested from the ambient and also discounts the energy spent in the node's operation.

2. Routing: The routing component has two roles. First, to exchange messages containing the battery level, aiming to find the lowest battery level of the available paths. Second, to select the path where the minimum energy node has the highest possible energy reserve among the available paths.

3. Data Aggregation: Knowing the lowest energy in the path, a data aggregation component regulates the traffic passing through each node. When this component aggregates data, the traffic load is reduced and the energy consumption on the path becomes smaller. Otherwise, the traffic load increases, which demands more energy from the nodes.

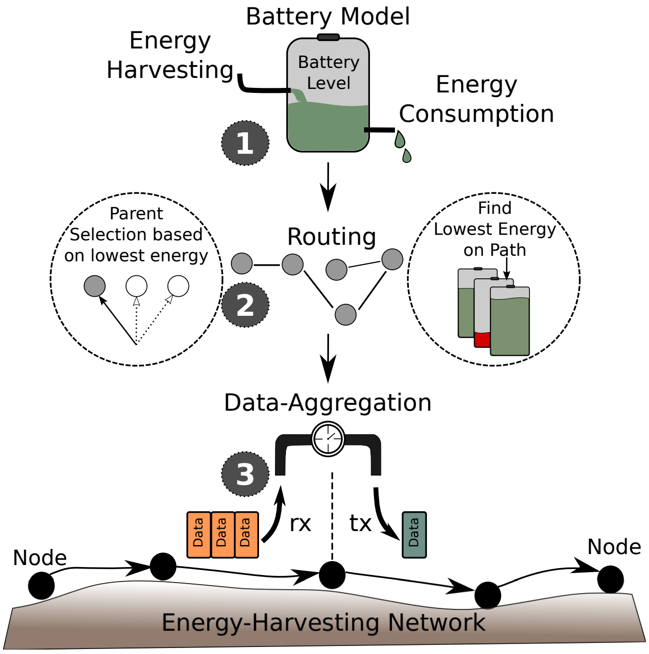


Figure 5.7: RAME overview.

Next, each RAME component is presented in detail.

5.3.2 Routing and Data Aggregation Solution for Operation of the Minimum Energy Node

This section presents the components of Routing and Aggregation for Minimum Energy (RAME), which is designed to dynamically set the aggregation level. Differently from the previous proposed solution, this mechanism has a module for estimating the residual energy in the batteries. Besides, RAME is designed to select paths with higher residual energy.

5.3.2.1 Kinetic Battery Model: Estimating the Node's Energy

A widely accepted solution to estimate the State-of-Charge (SoC) of a battery is the Kinetic Battery Model (KiBaM) [Manwell and McGowan, 1994]. This model considers the battery has two wells of charges. One is the available-charge well and the other is the bound-charge well, as can be seen on Figure 5.8.

The available-charge well supplies electrons to the output load, while the bound-charge well supplies electrons only to the available-charge well. The rate of charge that flows between the two wells is set by k, which is a fixed internal parameter, and also by the difference between h_1 and h_2. When $h_1 = 1$ the battery is fully charged and when $h_1 = 0$ the battery is fully discharged. Parameter

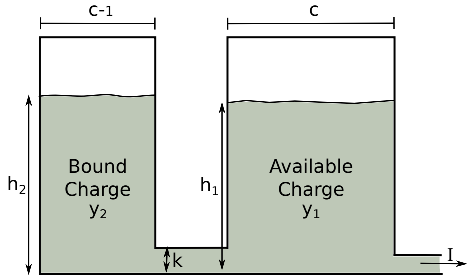


Figure 5.8: Kinetic Battery Model [Jongerden and Haverkort, 2009].

c is a capacity ratio and corresponds to the fraction of the total charge in the battery that is available.

Equations 5.4 and 5.5 show how to compute the amount of charge on the available and bound wells, which are denoted by y_1 and y_2, respectively [Jongerden and Haverkort, 2009].

$$y_1(t) = y_{1,0}e^{-k't} + \frac{(y_0k'c - I)(1 - e^{-k't})}{k'} - \frac{Ic(k't - 1 + e^{-k't})}{k'} \tag{5.4}$$

$$y_2(t) = y_{2,0}e^{-k't} + y_0(1 - c)(1 - e^{-k't}) - \frac{I(1 - c)(k't - 1 + e^{-k't})}{k'} \tag{5.5}$$

In these equations, k' is defined as $k' = k/c(1 - c)$, $y_{1,0}$ and $y_{2,0}$ are the amount of available and bound charges, respectively, at $t = 0$. In addition, the variable y_0 is the total charge at time $t = 0$, given by $y_0 = y_{1,0} + y_{2,0}$.

5.3.2.2 Routing: Finding the Lowest Energy and Selecting Parent

To find the minimum energy node, it is necessary to exchange control messages between the nodes. As the data-traffic flows from the nodes towards the sink, a particular node is interested in knowing the lowest energy level in its route to the sink. The idea is to know the minimum energy node of each possible route to the sink, and to select the one which has the highest energy. In that sense, this idea comes from the max-min optimization strategy.

Algorithm 5.2 presents how RAME determines the lowest energy in the route to the root. The procedure starts at the root node, broadcasting the energy estimation of its own battery. The nodes that have the root as parent will receive the $minE_p$ and will also transmit their own *my_minE*. Therefore, the other non-root nodes receive messages indicating the $minE_p$ for each parent p, and also select as preferred parent the one that has maximum $minE_p$. After selecting the preferred parent, the node computes and broadcasts *my_minE*, which is the minimum between *my_Energy* and *max*($minE_p$). The value stored in *my_minE* is sent to all candidate children, enabling the process to be repeated at all network nodes.

It is worth to note that Algorithm 5.2 does not address loop-avoidance or how to determine if a neighbor is a parent candidate, since these aspects are handled

Algorithm 5.2. Find Lowest Energy on Path and Parent Selection.

```
Initialize: candidate_set, my_Energy
Start
    for all parent p in candidate_set do
        Receive msg with $minE_p$
    end for
    Select p with max($minE_p$)
    my_minE ← min(my_Energy, max($minE_p$))
    Send msg with my_minE
End
```

by the routing protocol, such as RPL.

5.3.2.3 Dynamic Data Aggregation

The proposed data aggregation solution takes advantage of the small size of the M2M messages to enable the nodes to assemble messages with many payloads. A distinction should be made between the “unitary payload” and the “concatenated payload”. The former, also called small payload, refers to the smallest payload that can exist, while the latter means the payload formed by two or more unitary payloads. In this work, the unitary payload is considered to have a fixed size. Regarding data aggregation, RAME executes the following steps periodically:

1. **Extracting payloads:** It extracts the unitary payloads from all application messages received from the children, discarding the headers.
2. **Payload Aggregation and Concatenation:** It processes all unitary payloads, including the extracted and the produced by the node itself. There are two aggregation procedures for the gathered payloads: (i) aggregates sub-sets of unitary payloads into a single unitary payload using statistical functions (e.g. Maximum, Minimum, and Average), transforming multiple unitary payloads into a single one; and (ii) concatenates all the remaining unitary payloads from the previous aggregation procedure.
3. **Assembling payloads:** It creates a single application layer message containing the concatenated payloads.

The number of payloads of the final message is decided in step 2. Messages with a higher number of payloads will represent heavier traffic load for the nodes located on the route, especially for the node with lowest energy. To decide the number of unitary payloads inside the transmitted messages and consequently how many payloads must be aggregated, RAME uses the α parameter ($0 \leq \alpha \leq 1$), which is computed as a function of the lowest energy on the path. Equation 5.2 shows how α is used to compute the number of payloads transmitted.

$$Tx_{payloads} = (Rx_{payloads} * \alpha) + 1 \tag{5.2}$$

As α is the parameter that determines the number of payloads that will be transmitted, the aggregation level is given by $1 - \alpha$.

The next section details the implementation of RAME in an operating system and how the standard stack of M2M protocols have been extended.

5.3.3 Implementation in Contiki OS

This section presents how the different components of RAME have been implemented in Contiki OS. The reason to choose Contiki OS is the availability of the M2M protocols, such as CoAP, RPL, 6LowPAN, and IEEE 802.15.4.

5.3.3.1 Battery Model

The KiBaM model [Manwell and McGowan, 1994] was implemented in Contiki OS with the following modules: energy consumption and energy harvesting. The energy harvesting module of KiBaM reads a data-trace that contains the amount of harvested energy per minute. This module takes the data-trace as input, processes it and feeds the model with the equivalent charging current. The used energy harvesting dataset contains indoor radiant light measurements collected at Columbia University [Gorlatova et al., 2011]. More details about the harvested trace are given in Section 5.3.4. Regarding energy consumption, the battery model is based on Powertrace [Dunkels et al., 2011] functionalities. Therefore, KiBaM is able to measure the energy consumption in the following states: Transmit, Receive, Idle Listen, Active CPU, and Low Power CPU. The code that provides energy consumption and periodically computes the remaining energy in the battery is available [1].

5.3.3.2 Routing: Extended RPL

Regarding the routing aspects, the implementation of RAME has been developed as extension of the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL). RPL is a tree-based protocol that creates a Directed Acyclic Graph (DAG) to route the traffic. Standard RPL determines a neighbor node as parent candidate using the rank information. Rank is similar to hop-count, because for each hop in the route the rank increases a given amount. Since the primary objective of rank is to avoid loops in the route, additional metrics can be used to select the preferred parent among the parent candidates.

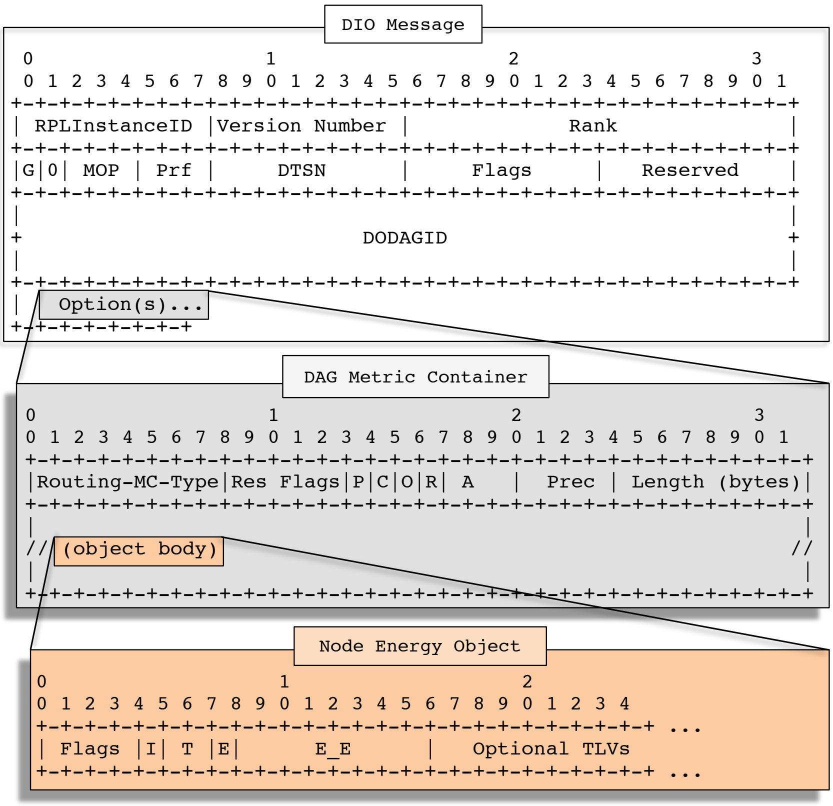
RPL has a set of ICMPv6 messages defined to exchange information among the nodes. One of them, named DAG Information Object (DIO), is very relevant for the RAME implementation. DIO is a RPL message that carries information about the RPL Instance and the configuration parameters of the DAG. DIO messages travel downwards in the network, which means that they go from the root to the nodes. However, the information inside DIO is used to compute the

---

[1] https://github.com/KineticBattery/Powertrace

upward paths. For that reason, RAME uses the DIO messages to determine the lowest energy level in the available paths.

Figure 5.9 shows the fields of a RAME DIO message with the DAG metric container and the Node Energy Object. In the DAG metric container, the field "Routing-MC-Type" is set to 2, which corresponds to the Energy Metric Container, according to RFC 6551 [Vasseur et al., 2012]. Besides, in the Node Energy Object, RAME's implementation sets the E bit as '1' in order to use the E_E 8-bit field of the message.



```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RPLInstanceID |Version Number |             Rank              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|G|0| MOP | Prf |     DTSN      |     Flags     |   Reserved    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                            DODAGID                            +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Option(s)...
+-+-+-+-+-+-+-+-+
```

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Routing-MC-Type|Res Flags|P|C|O|R| A   |  Prec | Length (bytes)|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
// (object body)                                               //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ ...
| Flags |I| T |E|      E_E      |  Optional TLVs
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ ...
```

Figure 5.9: DIO message and its fields.

Originally, E_E (Estimated-Energy) is an 8-bit unsigned integer field that indicates the estimated percentage of remaining energy. For the purpose of finding the lowest energy in the path, when a node running RAME sends a DIO message, the E_E field contains the numeric value stored in the variable *my_minE* (See Line 7 of Algorithm 5.2). In case a RAME node receives a DIO message from a parent candidate p, the E_E corresponds to the $minE_p$ (See Line 4 of Algorithm 5.2). Thus, using the E_E (Estimated-Energy) field of the DIO messages, it is possible to find the minimum energy level of the paths.

In addition, the RPL function that computes the cost of the parent candidates was adapted to use the minE_p contained in the received DIO messages. This change enables the RPL to compare the minE_p of all parent candidates, and to select the parent with highest minE_p.

5.3.3.3 Data Aggregation Integrated in the M2M Stack Protocol

The proposed Data Aggregation solution has been implemented to aggregate payloads of the Constrained Application Protocol (CoAP). The extraction of the CoAP payloads coming from children nodes is implemented on the IP layer. Thus, whenever a message passes through the IP layer, the extraction code verifies the fields of the message that identify the type of protocol. If this field indicates that it is a CoAP message, the code verifies if the message is self-produced or it is coming from neighbors.

By accessing the fields of the message that indicate the length of the non-CoAP header, it is possible to use the functionality available in Contki to parse the CoAP portion of the message. All payloads extracted are stored in a buffer. All unitary payloads have a fixed size of 2 bytes and the content of the payload is decoded as ”plain text”.

The CoAP implementation in Contiki OS allows clients to observe resources of the CoAP nodes, setting an interval in which the node will send data periodically. The implemented aggregation takes advantage of this functionality, since the aggregation has been developed to be triggered when the node has a new data observation to send to the client. Thus, the data aggregation procedures executed on the buffer take place when the node executes the code to send a new observation message. When the aggregation is triggered, the node determines an α value based on the minimum energy level on the path, and produces the CoAP message with one or multiple payloads.

The next section presents the evaluation and discusses the obtained results.

5.3.4 Performance Evaluation

This section describes the evaluation and the obtained results of RAME. Section 5.3.4.1 outlines the objectives of the evaluation. Section 5.3.4.2 presents the settings used for the evaluation and Section 5.3.4.3 shows the evaluation metrics. Section 5.3.4.4 presents the obtained results.

5.3.4.1 Objectives

The main objective of this evaluation is to show how the solutions regulate the energy consumption when the lowest energy node has different levels of stored energy, such as low, middle, and high state-of-charge. The expected result is to aggregate less data to use more energy from the battery when the lowest energy node approaches the full charge, and vice-versa.

5.3.4.2 Settings

Table 5.2 shows the used settings for KiBaM. The update interval is 5 minutes, which means that every 5 minutes the KiBaM updates the amount of energy charge. Regarding energy harvesting, this evaluation considers the dataset that contains indoor light energy measurements collected by the Columbia University [Gorlatova et al., 2011]. It provides a temporal-series of watt/cm^2, so to use these measurements as KiBaM input, the voltage is fixed at 5v, the area considered to harvest energy corresponds to 210 cm^2 and the conversion efficiency is 20%.

The hardware used is the following: MSP430 series 5, which has a MicroController Unit (MCU) of 16 bits with 16kB internal RAM and 128kB Flash. The transceiver is TI CC2520 (2.4GHz), compatible with IEEE 802.15.4 and IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN).

To measure the energy consumption, the Powertrace tool, which is available in Contiki OS, has been used. Powertrace is able to measure the energy consumption in the following consumption states: (i) Active CPU, (ii) Low Power Mode CPU, (iii) Transmit, (iv) Listen, (v) Idle Listen. In addition, Powertrace is fully integrated with the above mentioned hardware, which means that it is able to measure the energy activities related not only to the data-traffic communication but also to the data-processing and control activities demanded by the protocols. Some additional settings are shown in Table 5.2.

Table 5.2: Battery and Communication settings.

Configuration (i)	**Info**
Battery Capacity (B_c)	1000000 microAh
Max Bound Charge	10% of B_c
Update Periodicity	5 minutes
Application Protocol	CoAP
IP layer	6LowPAN
Low Duty Cycling	ContikiMac
Mac and Physical	IEEE 802.15.4
Max Available Charge	90% of B_c
Internal rate K	0.1
Unitary payload size	2 Bytes
CoAP Obs Interval	5 minutes
Routing Protocol	RPL
Wakeup Frequency	2Hz
Wireless Range	20 m

Some preliminary tests have been conducted to select the best function for the alpha parameter (see Equation 5.2). Figure 5.10 shows three different functions used to determine the α parameter according to the lowest energy in the path. For the purpose of this evaluation, the polynomial function has been used in all experiments.

After being configured, Contiki OS is compiled and the binary image can be emulated using Cooja, which also enables the emulated nodes to be simulated as a wireless network. The obtained results are based on the Cooja simulation. Since the simulations are very time-consuming, the network was simulated with 40 nodes and the number of disjoint paths is fixed in 4. The nodes are located with 10 meters of spacing between each other. At the beginning of the simula-

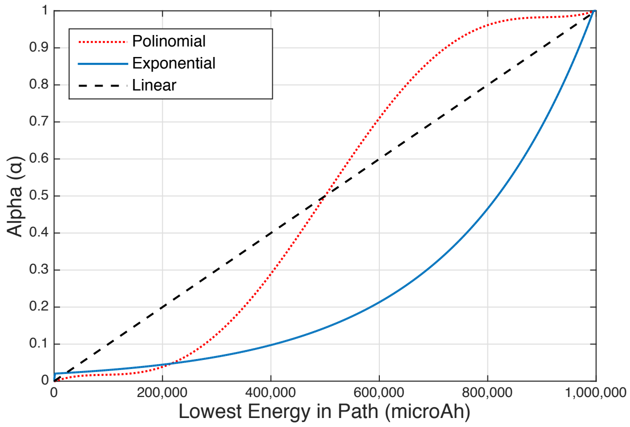


Figure 5.10: Alpha (α) is a function of the lowest energy level.

tion, the minimum energy node of each disjoint path begins the simulation with 5% less energy than the other nodes.

5.3.4.3 Evaluated Solutions and Metrics

Two solutions have been evaluated, namely: Routing and Aggregation for Minimum Energy (RAME) and the standard M2M protocols. The standard solution does not perform any data aggregation and uses the Minimum Rank with Hysteresis Objective Function as metric for parent selection [Gnawali and Levis, 2012].

The following metrics have been used to measure the performance of RAME and the standard protocols.

- Energy Consumption of the Lowest Energy Node: This metric corresponds to the sum of the energy consumed by the minimum energy nodes divided by the number of disjoint paths. The consumption values have three categories: CPU, Transmission (Tx), and Reception (Rx). The reception category considers the effective reception of data and the idle listen state.
- Residual Energy of the Lowest Energy Node and Linear Trend: It measures over time the amount of charge that remains in the battery of the minimum energy node. In order to facilitate the interpretation of the tendency of the residual energy, a linear regression approach is applied on the obtained residual energy data to find a linear trend.

5.3.4.4 Results

Figure 5.11 shows the average energy consumed by the nodes with lowest energy in the disjoint paths. As can be noticed, in comparison to the standard version of the M2M protocols, RAME reduces the energy consumption in more than 12%. The gain is caused mostly by the concatenation of payloads, since for

$\alpha = 1$ there is no aggregation of payloads. Therefore, for these results, both solutions delivered the same amount of payloads. It means that the gain in terms of energy will be even higher when RAME uses $\alpha = 0$.

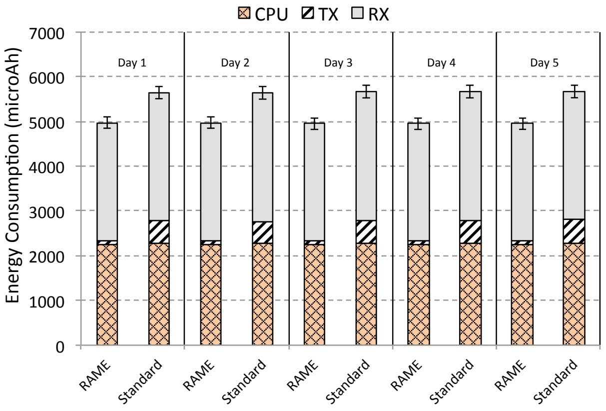


Figure 5.11: Average energy consumed by the node with lowest energy, considering $\alpha = 1$.

Figure 5.12 presents how the residual energy of the minimum energy node changes over 5 days, starting the batteries with 85% (High State-of-Charge). Both RAME and the standard solution have a decreasing linear tendency, indicating that the energy consumption is higher than the harvested energy. At this level of battery, it is desirable to have a decreasing residual tendency, since the battery only has about 15% of capacity left to store more energy. At this point, an increasing energy level could cause battery overflow, which would waste energy.

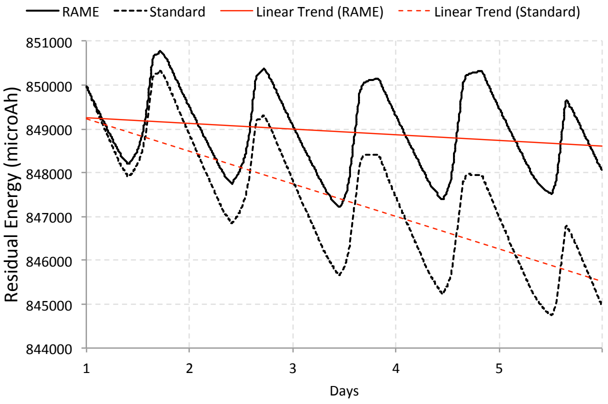


Figure 5.12: Battery initiates with 85%.

Figure 5.13 shows the residual energy in a scenario where the lowest energy node has 45% of battery (Intermediate State-of-Charge). This figure also presents the linear trend of RAME, which is almost a constant line. At this level of residual energy, the level of data aggregation is around 0.5, which means that half of the

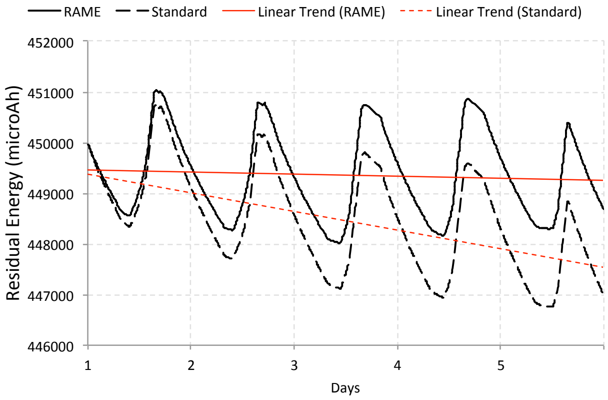


Figure 5.13: Battery begins with 45%.

received payloads are aggregated before transmission. On the other hand, the standard M2M protocols maintain a higher energy consumption rate.

Another case is when the lowest energy node has 15% of battery (Low State-of-Charge). Figure 5.14 shows that RAME has an increasing linear trend, which means that over time it is accumulating energy on the battery of the lowest energy node. However, the standard solution does not have a positive linear trend, since the energy consumed by this solution is significantly higher.

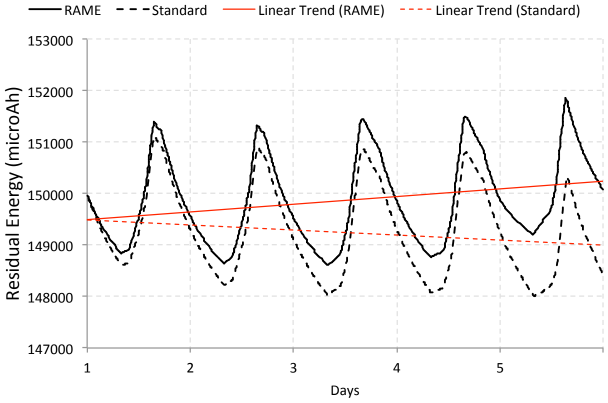


Figure 5.14: Battery starts at 15%.

A factor that contributes for the better performance of RAME is that it selects the paths where the minimum energy node has the highest possible energy reserve.

5.4 Summary

This chapter addressed the problem of how to maintain nodes at neutral operation, as well as reducing the congestion caused by communication overhead. To solve these problems, two mechanisms have been proposed. Both solutions regulate the energy consumption by dynamically determining an aggregation level that is used to control the amount of traffic flow through the nodes.

The first mechanism, called Data Aggregation for energy harVesting NETworks (DAV-NET), is presented in Section 5.2 and regulates the energy consumption of the nodes using as main parameter for the aggregation the residual energy in the battery. The design and evaluation of this solution showed some drawbacks. For instance, DAV-NET does not consider the current conditions of the neighbor nodes when a node decides the amount of data that must be aggregated.

With the insight provided by DAV-NET, Section 5.3 introduced Routing and Aggregation for Minimum Energy (RAME). This solution tackled the drawbacks of DAV-NET, since it considers the current conditions of the nodes composing the path towards the gateway. In summary, RAME exchanges information to determine the lowest battery reserve in the path, and based on this information the nodes define the aggregation level to be applied on the network traffic. In addition, RAME is implemented in a real system for low-power nodes, including the extension and adaptation of standard M2M protocols, such as CoAP and RPL.

The next chapter presents the conclusions and future works.

Chapter 6

Conclusions and Future Work

> I concluded that even this endeavor is like trying to chase the wind!
>
> *(King Solomon)*

Contents

In recent years, relevant progress has been made in information and communication technology. Numerous innovative applications have been emerged due to this progress. Machine-to-machine communications have contributed to this trend by unlocking several opportunities related to monitoring and remote control applications, namely smart grid, smart cities, transportation, and e-health. Despite numerous efforts on this field, including the ones promoted by the standardization bodies, some important open issues continue to exist. This thesis proposed several mechanisms to progress the current state-of-the-art by improving the energy efficiency of group communication in networks powered by traditional battery and by renewable energy sources. This chapter recaps the covered issues and highlight the major contributions successfully achieved. Besides, it ends showing the envisioned future research of this work.

6.1 Synthesis of the Thesis

Machine-to-machine presents a large potential to lead the information and technology industry towards an era of intelligent networked applications. However, the real progress of M2M communications depends on a myriad of complex factors. In special, this thesis has identified and proposed solutions to advance the energy efficiency of M2M group communication, considering different metrics for energy efficiency and group communication in several scenarios.

This thesis, in Chapter 2, covers the background concepts and definitions of machine-to-machine environments, including the types of functionalities executed by the nodes and the features of the smart devices. This chapter also details and gives a perspective of the main machine-to-machine scenarios and applications, and the types of traffic produced by machine-to-machine devices. Another important topic outlined in this chapter is the discussion about IETF standards defined for M2M communication, including details about the CoAP the RPL protocols.

Chapter 3 presents a comprehensive analysis and discussion about the challenges, requirements, state-of-the-art, and open issues. The three challenges presented are related to energy efficiency, group communication and reducing traffic for M2M scenarios. This chapter narrowed the vast set of M2M applications to allow the identification of communication requirements, presenting applications related to the challenges. After the discussion around the challenges and identification of the requirements, this chapter presents a holistic state-of-the-art study, with the aim to show how the existing literature solutions solve or fulfill the selected communication requirements. To accomplish the objective of this chapter, it presents an analysis about the open issues and research directions. To be specific, two research directions have been followed, both aiming to improve energy efficiency and M2M group communication, one is focused on nodes that have limited energy reserves and the other is centered on nodes that have renewable sources of energy.

Chapter 4 presents an adaptable and general approach for M2M group communications, called Two Tier Aggregation (TTA), that reduces overhead and eliminates data redundancy. As TTA is a general approach, it does not show how to find a network flow that will produce the best network lifetime performance. To solve the aspects related to the network flow selection, two heuristics were designed. After presenting the heuristics and their performances, this chapter introduces a mathematical model, formulated as an Integer Linear Program, with the objective of finding the optimal network flow solution for TTA that maximizes the network lifetime. Besides, a performance comparison between the heuristics and the optimal or near-optimal solutions is presented. This chapter ends with the presentation of the implementation of a real prototype and testbed experiments. The experiments showed the real gain, in terms of energy consumption, of the proposed TTA. Besides, the obtained results highlighted that the aggregation cost is not negligible and, therefore, how it can be considered as a criteria to make the aggregation more efficient.

Chapter 5 addresses the problem of how to maintain nodes with recharging capabilities (i.e. solar, wind, vibration) at neutral operation. As the data communication is critical for energy consumption, the proposed mechanisms regulate the energy consumption by dynamically determining the amount of traffic that flows through the nodes. The first mechanism, called Data Aggregation for energy harVesting NETworks (DAV-NET), does the regulation using as main parameter the residual energy in the battery. The design and evaluation of this solution showed some drawbacks. For instance, DAV-NET does not consider the current conditions of the neighbor nodes when a node decides the amount of data that must be aggregated. With the insights provided by DAV-NET, this chapter introduces the Routing and Aggregation for Minimum Energy (RAME). This solution improves the drawbacks of DAV-NET, since it considers the current conditions of the nodes composing the path towards the gateway. In addition, RAME is more complete than DAV-NET because RAME was implemented in a real system for low-power nodes. In addition, RAME was designed as an extension of standard M2M protocols, such as CoAP and RPL. Moreover, RAME was evaluated using a tool that enables the emulation of a real board with accurate energy consumption model.

6.2 Contributions

The defined goal of this thesis, shown in Chapter 1, is to improve energy efficiency of machine-to-machine group communication. The first specific goal is to define a generic approach suitable and adaptable for machine-to-machine group communication to eliminate data redundancy and reduce overhead. Another objective is to propose a set of group communication mechanisms that aim to improve the energy efficiency in traditional battery-powered with fixed energy budget and renewable energy sources. The third objective is to study the performance of the proposed mechanisms in various evaluation methods, including simulation, mathematical models, and real experiments.

These goals have tailored the work presented in this thesis and have resulted in the following contributions:

Contribution 1, A generic approach for M2M group communication

> This is a transversal contribution, presented in Chapter 4, Section 4.2, since it is present in all mechanisms, integrally or partially, to improve M2M group communication. The proposed approach is adaptable to the existing groups on the network and relies on two distinct types of processing functions to enable less overhead and eliminate data redundancy. This approach, Two Tier Aggregation, is flexible enough to be used on different group communication scenarios and allows the groups to have particular communication settings;

Contribution 2, Heuristics to prolong the network lifetime

> Routing heuristics are the natural complementary step in order to apply the proposed Two Tier Aggregation. The proposed heuristics, Chapter 4, Section 4.3, are an important contribution because they decide aspects related to the points of aggregation and how traffic will flow over the network. Without well designed heuristics, TTA performs poorly in terms of network lifetime. Thus, the proposed heuristics are tree-based solutions that find compatible groups in terms of communication periodicity to reduce the overhead and eliminate the data redundancy;

Contribution 3, Model formulation for network lifetime maximization

Heuristics have the benefit of requiring low computational resources and are easily turned in distributed approaches. However, heuristics can have a low performance in comparison to the optimal. The formulation of mathematical models is a powerful technique able to find the optimal solution for a given problem. Even though the computation cost to find the optimal solution is too high or complex to be implemented, the optimal solution is important for the study of a problem because it defines an upper bound. Motivated by these factors, Chapter 4, Section 4.4, shows the formulation of the network lifetime upper bound obtained using Integer Linear Programming. In addition, a comparison between the optimal and the heuristics solutions is presented;

Contribution 4, Prototyping and testbed experimentation

This contribution is related to the implementation and evaluation tasks in real devices, presented in Chapter 4, Section 4.5. The experimentation using prototypes and real environments poses several challenges not commonly considered in simulation or mathematical models. The real devices enable a realistic notion of the gains and limitations of the proposed solution. The conducted testbed experimentation has focused on measuring the energy consumption, specifically testing basic cases and analyzing the impact the aggregation procedures have on the energy;

Contribution 5, Mechanisms for energy sustainable operation

This contribution, presented in Chapter 5, includes the design and evaluation of mechanisms to keep in sustainable operation devices with recharging energy sources. The focus of these mechanisms was to regulate the amount of traffic flowing through a node, using dynamic data aggregation procedures. In the first proposed mechanism, each node regulates autonomously its traffic using as main parameter the residual energy in the reserve. In the second mechanism, each node decides the amount of traffic to be aggregated by considering the minimum energy level in the path towards the gateway;

Contribution 6, Adaptation and extension of standard M2M protocol stack

This contribution, shown in Chapter 5, Section 5.3.3, is related to the fact that the standard M2M protocols, such as CoAP and RPL, have been adapted and extended to promote the convergence and the advance of the M2M protocols in energy harvesting scenarios;

The following subsection outlines research directions for possible future works.

6.3 Future Work

Group communication mechanisms proposed and evaluated in this thesis aim to improve the energy efficiency of the M2M communications. The obtained results through simulation, mathematical models, and real experiments show that the proposed mechanisms outperforms the traditional solutions. However, there are some open issues that could be addressed in the future, such as the consideration of event-based instead of periodic communication and the formulation of a distributed optimal solution for the Two Tier Aggregation.

An aspect that could be tackled is the adoption of a reliable mechanism to protect aggregated messages. It would be an important topic to be explored because the aggregation procedures used in TTA create an incompatibility with ACK-based retransmissions approaches. This incompatibility exists because the the client cannot identify the data producers nodes in the aggregated message. The proposal of a mechanism to improve reliability of aggregated message could use information to estimate the loss condition of the network to sent replicas of the aggregated message. Other approach could be to use a flag in the aggregated message header to indicate the ACK receiver node and preventing the message to be aggregated by other nodes.

Another research direction to be explored is how the proposed solutions can be adapted and integrated in 5th Generation Networks (5G). Much of the group management is envisioned to be executed by the M2M gateway and functionalities of this device need to be integrated in 5G networks. This integration must be done considering that the M2M devices can cause congestion due to the large number of connections of small data transmission. In addition to the integration aspects, management of the groups is also a relevant topic to be advanced, since it refers to the communication settings, the firmware updates, mobility of the group members, and the control of membership.

Finally, neutral operation of nodes with energy harvesting capabilities is a research direction that still needs progress. Most of the current standard protocols for constrained devices rely on the premise that the energy consumption must be reduced at most. However, this premise is not always valid for nodes able to recharge their batteries using the ambient. Thus, most of the protocols must be rethought under the energy harvesting possibility. The future energy harvesting solutions must also consider that some legacy devices, using traditional batteries, may be present. In addition, in this research direction, wearable devices represent a prolific field for future revolutionary applications, mainly for health applications.

Bibliography

[(3GPP), 2010] (3GPP), T. G. P. P. (2010). Service requirements for machine-type-communication (mtc).

[Access, 2007] Access, E. U. T. R. (2007). Long term evolution (lte) physical layer; general description. In *Draft1) Report of the 35th 3GPP TSG RAN meeting*.

[Akpakwu et al., 2018] Akpakwu, G. A., Silva, B. J., Hancke, G. P., and Abu-Mahfouz, A. M. (2018). A survey on 5g networks for the internet of things: communication technologies and challenges. *IEEE Access*, 6:3619–3647.

[Al-Karaki et al., 2009] Al-Karaki, J. N., Ul-Mustafa, R., and Kamal, A. E. (2009). Data aggregation and routing in wireless sensor networks: Optimal and heuristic algorithms. *Computer networks*, 53(7):945–960.

[Ali et al., 2015] Ali, A., Hamouda, W., and Uysal, M. (2015). Next generation m2m cellular networks: challenges and practical considerations. *IEEE Communications Magazine*, 53(9):18–24.

[Alliance, 2007] Alliance, W. (2007). Wi-fi standards.

[Alliance, 2006] Alliance, Z. (2006). Zigbee specification. *Document 053474r06, Version*, 1.

[Ashrafuzzaman and Fapojuwo, 2018] Ashrafuzzaman, K. and Fapojuwo, A. O. (2018). Efficient and agile carrier sense multiple access in capillary machine-to-machine communication networks. *IEEE Access*.

[Atmel, 2016] Atmel (2016). Atmega128, accessed:.

[Azari and Miao, 2017] Azari, A. and Miao, G. (2017). Network lifetime maximization for cellular-based m2m networks. *IEEE Access*, 5:18927–18940.

[Banks and Gupta, 2015] Banks, A. and Gupta, R. (2015). Message queuing telemetry transport.

[Beale and Morioka, 2011] Beale, M. and Morioka, Y. (2011). Wireless machine-to-machine communication. In *Microwave Conference (EuMC), 2011 41st European*, pages 115 –118.

[Bicakci and Tavli, 2010] Bicakci, K. and Tavli, B. (2010). Prolonging network lifetime with multi-domain cooperation strategies in wireless sensor networks. *Ad Hoc Networks*, 8(6):582–596.

[Bluetooth, 2001] Bluetooth, S. (2001). Bluetooth specification version 1.1. *Available HTTP: http://www. bluetooth. com.*

[Boccardi et al., 2014] Boccardi, F., Heath, R. W., Lozano, A., Marzetta, T. L., and Popovski, P. (2014). Five disruptive technology directions for 5g. *IEEE Communications Magazine*, 52(2):74–80.

[Booysen et al., 2012] Booysen, M. J., Gilmore, J. S., Zeadally, S., and Van Rooyen, G.-J. (2012). Machine-to-machine (m2m) communications in vehicular networks.

[Bradley et al., 1977] Bradley, S., Hax, A., and Magnanti, T. (1977). *Applied mathematical programming*. Addison Wesley.

[Casilari et al., 2010] Casilari, E., Cano-García, J. M., and Campos-Garrido, G. (2010). Modeling of current consumption in 802.15. 4/zigbee sensor motes. *Sensors*, 10(6):5443–5468.

[Choi and Chung, 2013] Choi, D.-W. and Chung, C.-W. (2013). REQUEST+: A framework for efficient processing of region-based queries in sensor networks. *Information Sciences*, 248:151–167.

[Cisco, 2016] Cisco, C. V. N. I. (2016). Global mobile data traffic forecast update, 2015–2020 white paper, 2016. *Online: https://www. cisco. com/c/dam/m/en_in/innovation/enterprise/assets/mobile-white-paper-c11-520862. pdf.*

[Di Stasi et al., 2014] Di Stasi, G., Karlsson, J., Avallone, S., Canonico, R., Kassler, A., and Brunstrom, A. (2014). Combining multi-path forwarding and packet aggregation for improved network performance in wireless mesh networks. *Computer Networks*, 64:26–37.

[Dunkels et al., 2011] Dunkels, A., Eriksson, J., Finne, N., and Tsiftes, N. (2011). Powertrace: Network-level power profiling for low-power wireless networks.

[Eu and Tan, 2012] Eu, Z. A. and Tan, H.-P. (2012). Adaptive opportunistic routing protocol for energy harvesting wireless sensor networks. In *Communications (ICC), 2012 IEEE International Conference on*, pages 318–322. IEEE.

[Fadlullah et al., 2011] Fadlullah, Z., Fouda, M., Kato, N., Takeuchi, A., Iwasaki, N., and Nozaki, Y. (2011). Toward intelligent m2m communications in smart grid. *IEEE Communications Magazine*, 49(4):60–65.

[Fasolo et al., 2007] Fasolo, E., Rossi, M., Widmer, J., and Zorzi, M. (2007). In-network aggregation techniques for wireless sensor networks: a survey. *Wireless Communications, IEEE*, 14(2):70–87.

[Fielding et al., 1999] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and Berners-Lee, T. (1999). Hypertext transfer protocol–http/1.1.

[Gao et al., 2014] Gao, D., Liu, Y., Zhang, F., and Song, J. (2014). Data aggregation routing for rechargeable wireless sensor networks in forest monitoring. *Wireless Personal Communications*, 79(1):773–788.

[Ghavimi and Chen, 2015] Ghavimi, F. and Chen, H.-H. (2015). M2m communications in 3gpp lte/lte-a networks: Architectures, service requirements, challenges, and applications. *IEEE Communications Surveys & Tutorials*, 17(2):525–549.

[Gnawali and Levis, 2012] Gnawali, O. and Levis, P. (2012). The minimum rank with hysteresis objective function.

[Gorlatova et al., 2011] Gorlatova, M., Zapas, M., Xu, E., Bahlke, M., Kymissis, I. J., and Zussman, G. (2011). CRAWDAD dataset columbia/enhants (v. 2011-04-07). Downloaded from `http://crawdad.org/columbia/enhants/20110407`.

[Gregori and Payaró, 2012] Gregori, M. and Payaró, M. (2012). Energy-efficient transmission for wireless energy harvesting nodes. *arXiv preprint arXiv:1212.1115*.

[Group et al., 2004] Group, I. . W. et al. (2004). Ieee standard for local and metropolitan area networks-part 16: Air interface for fixed broad-band wireless access systems. *IEEE Std. 802.16-2004*.

[Gunduz et al., 2014] Gunduz, D., Stamatiou, K., Michelusi, N., and Zorzi, M. (2014). Designing intelligent energy harvesting communication systems. *IEEE Communications Magazine*, 52(1):210–216.

[Handy et al., 2002] Handy, M., Haase, M., and Timmermann, D. (2002). 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.

[He et al., 2015] He, Y., Cheng, X., Peng, W., and Stuber, G. L. (2015). A survey of energy harvesting communications: Models and offline optimal policies. *IEEE Communications Magazine*, 53(6):79–85.

[Heinzelman et al., 2002] Heinzelman, W. B., Chandrakasan, A. P., and Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *Wireless Communications, IEEE Transactions on*, 1(4):660–670.

[Hua and Yum, 2008] Hua, C. and Yum, T.-S. P. (2008). Optimal routing and data aggregation for maximizing lifetime of wireless sensor networks. *IEEE/ACM Transactions on networking*, 16(4):892–903.

[Hurni et al., 2011] Hurni, P., Nyffenegger, B., Braun, T., and Hergenroeder, A. (2011). On the accuracy of software-based energy estimation techniques. In Marron, P. J. and Whitehouse, K., editors, *Wireless Sensor Networks: 8th European Conference, EWSN 2011, Bonn, Germany, February 23-25, 2011. Proceedings*, pages 49–64, Berlin, Heidelberg. Springer Berlin Heidelberg.

[Hussain and Islam, 2007] Hussain, S. and Islam, O. (2007). An energy efficient spanning tree based multi-hop routing in wireless sensor networks. In *Wireless Communications and Networking Conference, 2007.*, pages 4383–4388. IEEE.

[Hussein and Batarseh, 2011] Hussein, A. A.-H. and Batarseh, I. (2011). A review of charging algorithms for nickel and lithium battery chargers. *IEEE Transactions on Vehicular Technology*, 60(3):830–838.

[IEEE, 2016] IEEE (2016). Ieee standard for low-rate wireless networks. *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)*, pages 1–709.

[Intanagonwiwat et al., 2003] Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., and Silva, F. (2003). Directed diffusion for wireless sensor networking. *Networking, IEEE/ACM Transactions on*, 11(1):2–16.

[Ishaq et al., 2016] Ishaq, I., Hoebeke, J., Moerman, I., and Demeester, P. (2016). Observing coap groups efficiently. *Ad Hoc Networks*, 37:368–388.

[Ishaq et al., 2014] Ishaq, I., Hoebeke, J., Van den Abeele, F., Rossey, J., Moerman, I., and Demeester, P. (2014). Flexible Unicast-Based Group Communication for CoAP-Enabled Devices. *Sensors*, 14(6):9833–9877.

[Jeong et al., 2016] Jeong, S., Kim, H., Noh, D. K., and Yoon, I. (2016). Energy-aware data aggregation scheme for energy-harvesting wireless sensor networks. In *Computer Communication and the Internet (ICCCI), 2016 IEEE International Conference on*, pages 140–143. IEEE.

[Jesus et al., 2015] Jesus, P., Baquero, C., and Almeida, P. S. (2015). A survey of distributed data aggregation algorithms. *IEEE Communications Surveys & Tutorials*, 17(1):381–404.

[Jongerden and Haverkort, 2009] Jongerden, M. R. and Haverkort, B. R. (2009). Which battery model to use? *IET software*, 3(6):445–457.

[Kacimi et al., 2013] Kacimi, R., Dhaou, R., and Beylot, A.-L. (2013). Load balancing techniques for lifetime maximizing in wireless sensor networks . *Ad Hoc Networks*, 11(8):2172 – 2186.

[Kalpakis et al., 2003] Kalpakis, K., Dasgupta, K., and Namjoshi, P. (2003). Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks. *Computer Networks*, 42(6):697–716.

[Karakus et al., 2013] Karakus, C., Gurbuz, A., and Tavli, B. (2013). Analysis of energy efficiency of compressive sensing in wireless sensor networks. *IEEE Sensors Journal*, 13(5):1999–2008.

[Katsaros et al., 2014] Katsaros, K., Chai, W., Wang, N., Pavlou, G., Bontius, H., and Paolone, M. (2014). Information-centric networking for machine-to-machine data delivery: a case study in smart grid applications. *IEEE Network*, 28(3):58–64.

[Kim et al., 2017] Kim, D. M., Sorensen, R. B., Mahmood, K., Osterbo, O. N., Zanella, A., and Popovski, P. (2017). Data aggregation and packet bundling of uplink small packets for monitoring applications in lte. *IEEE Network*, 31(6):32–38.

[Kleerekoper and Filer, 2014] Kleerekoper, A. and Filer, N. (2014). DECOR:

Distributed construction of load balanced routing trees for many to one sensor networks . *Ad Hoc Networks*, 16(0):225 – 236.

[Lattanzi et al., 2007] Lattanzi, E., Regini, E., Acquaviva, A., and Bogliolo, A. (2007). Energetic sustainability of routing algorithms for energy-harvesting wsn. *Computer Communications*, 30(14):2976–2986.

[Lee et al., 2010] Lee, C.-H., Chung, C.-W., and Chun, S.-J. (2010). Effective processing of continuous group-by aggregate queries in sensor networks. *Journal of Systems and Software*, 83(12):2627–2641.

[Li et al., 2006] Li, D., Cao, J., Liu, M., and Zheng, Y. (2006). Construction of optimal data aggregation trees for wireless sensor networks. In *Computer Communications and Networks, 2006. ICCCN 2006. Proceedings. 15th International Conference on*, pages 475–480. IEEE.

[Lin and Chen, 2017] Lin, H.-C. and Chen, W.-Y. (2017). An approximation algorithm for the maximum-lifetime data aggregation tree problem in wireless sensor networks. *IEEE Transactions on Wireless Communications*, 16(6):3787–3798.

[Lin et al., 2010] Lin, H.-C., Li, F.-J., and Wang, K.-Y. (2010). Constructing maximum-lifetime data gathering trees in sensor networks with data aggregation. In *Communications (ICC), 2010 IEEE International Conference on*, pages 1–6. IEEE.

[Lindsey and Raghavendra, 2002] Lindsey, S. and Raghavendra, C. S. (2002). PEGASIS: Power-efficient gathering in sensor information systems. In *Aerospace conference proceedings, 2002. IEEE*, volume 3, pages 3–1125. IEEE.

[Lioumpas et al., 2011] Lioumpas, A., Alexiou, A., Anton-Haro, C., and Navaratnam, P. (2011). Expanding lte for devices: Requirements, deployment phases and target scenarios. *Wireless Conference 2011 - Sustainable Wireless Technologies (European Wireless), 11th European*, pages 1 –6.

[Liu et al., 2014] Liu, C. H., Yang, B., and Liu, T. (2014). Efficient naming, addressing and profile services in Internet-of-Things sensory environments . *Ad Hoc Networks*, 18(0):85 – 101.

[Lo et al., 2013] Lo, A., Law, Y., and Jacobsson, M. (2013). A cellular-centric service architecture for machine-to-machine (m2m) communications. *IEEE Wireless Communications*, 20(5):143–151.

[Long et al., 2017] Long, J., Dong, M., Ota, K., and Liu, A. (2017). A green tdma scheduling algorithm for prolonging lifetime in wireless sensor networks. *IEEE Systems Journal*, 11(2):868–877.

[Luo et al., 2011] Luo, D., Zhu, X., Wu, X., and Chen, G. (2011). Maximizing lifetime for the shortest path aggregation tree in wireless sensor networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 1566–1574. IEEE.

[Ma et al., 2009] Ma, J., Lou, W., Wu, Y., Li, M., and Chen, G. (2009). Energy efficient TDMA sleep scheduling in wireless sensor networks. In *INFOCOM 2009*, pages 630–638. IEEE.

[Madden et al., 2002] Madden, S., Franklin, M. J., Hellerstein, J. M., and Hong, W. (2002). Tag: A tiny aggregation service for ad-hoc sensor networks. *ACM SIGOPS Operating Systems Review*, 36(SI):131–146.

[Manwell and McGowan, 1994] Manwell, J. and McGowan, J. (1994). Extension of the kinetic battery model for wind/hybrid power systems. In *EWEC*, pages 284–289.

[Marsch et al., 2012] Marsch, P., Raaf, B., Szufarska, A., Mogensen, P., Guan, H., Farber, M., Redana, S., Pedersen, K., and Kolding, T. (2012). Future mobile communication networks: Challenges in the design and operation. *Vehicular Technology Magazine, IEEE*, 7(1):16 –23.

[Matoba et al., 2011] Matoba, K., Abiru, K., and Ishihara, T. (2011). Service oriented network architecture for scalable m2m and sensor network services. In *15th International Conference on Intelligence in Next Generation Networks (ICIN)*, pages 35 –40.

[Mehmood et al., 2017] Mehmood, Y., Haider, N., Imran, M., Timm-Giel, A., and Guizani, M. (2017). M2m communications in 5g: state-of-the-art architecture, recent advances, and research challenges. *IEEE Communications Magazine*, 55(9):194–201.

[Mottola and Picco, 2011] Mottola, L. and Picco, G. P. (2011). Muster: Adaptive energy-aware multisink routing in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 10(12):1694–1709.

[Naeem et al., 2017] Naeem, M., Ejaz, W., Karim, L., Ahmed, S. H., Anpalagan, A., Jo, M., and Song, H. (2017). Distributed gateway selection for m2m communication in cognitive 5g networks. *IEEE Network*, 31(6):94–100.

[Ozel et al., 2015] Ozel, O., Tutuncuoglu, K., Ulukus, S., and Yener, A. (2015). Fundamental limits of energy harvesting communications. *IEEE Communications Magazine*, 53(4):126–132.

[Pandey et al., 2011] Pandey, S., Kim, M.-S., Choi, M.-J., and Hong, J. (2011). Towards management of machine to machine networks. In *Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific*, pages 1 –7.

[Peng and Low, 2013] Peng, S. and Low, C. P. (2013). Energy neutral routing for energy harvesting wireless sensor networks. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 2063–2067. IEEE.

[Prim, 1957] Prim, R. C. (1957). Shortest connection networks and some generalizations. *Bell System Technical Journal*, 36(6):1389–1401.

[Riker et al., 2014a] Riker, A., Cerqueira, E., Curado, M., and Monteiro, E. (2014a). Data aggregation for group communication in machine-to-machine environments. In *Wireless Days (WD), 2014 IFIP*, pages 1–7. IEEE.

[Riker et al., 2015a] Riker, A., Cerqueira, E., Curado, M., and Monteiro, E. (2015a). Controlling data aggregation and communication periodicity in energy harvesting networks. In *IEEE Online Conference onGreen Communications (OnlineGreenComm)*, pages 72–77. IEEE.

[Riker et al., 2015b] Riker, A., Cerqueira, E., Curado, M., and Monteiro, E. (2015b). Data aggregation for machine-to-machine communication with energy harvesting. In *IEEE International Workshop on Measurements & Networking (M&N)*, pages 1–6. IEEE.

[Riker et al., 2016a] Riker, A., Cerqueira, E., Curado, M., and Monteiro., E. (2016a). A two-tier adaptive data aggregation approach for m2m group-communication. *IEEE Sensors Journal*, 16(3):823–835.

[Riker et al., 2014b] Riker, A., Cruz, T., Marques, B., Curado, M., Simoes, P., and Monteiro, E. (2014b). Efficient and secure m2m communications for smart metering. In *Emerging Technology and Factory Automation (ETFA), 2014 IEEE*, pages 1–7. IEEE.

[Riker et al., 2014c] Riker, A., Curado, M., and Monteiro, E. (2014c). Group communication in machine-to-machine environments. In *Wireless Networking for Moving Objects*, pages 219–238. Springer.

[Riker et al., 2016b] Riker, A., Curado, M., and Monteiro, E. (2016b). Role coordination in large-scale and highly-dense internet-of-things. In *Internet of Things (IoT) in 5G Mobile Technologies*, pages 77–98. Springer.

[Riker et al., 2017a] Riker, A., Curado, M., and Monteiro, E. (2017a). Neutral operation of the minimum energy node in energy-harvesting environments. In *IEEE Symposium on Computers and Communications (ISCC)*, pages 477–482. IEEE.

[Riker et al., 2017b] Riker, A., Fonseca, C. M., Curado, M., and Monteiro, E. (2017b). Energy-efficient multigroup communication. *Transactions on Emerging Telecommunications Technologies.*

[Riker et al., 2013] Riker, A., Granjal, J., Curado, M., and Monteiro, E. (2013). Middleware group communication mechanisms in m2m environments. In *2nd Joint ERCIM eMobility and MobiSense Workshop*, page 46.

[Shafiq et al., 2012] Shafiq, M. Z., Ji, L., Liu, A. X., Pang, J., and Wang, J. (2012). A first look at cellular machine-to-machine traffic: large scale measurement and characterization. In *ACM SIGMETRICS Performance Evaluation Review*, volume 40, pages 65–76. ACM.

[Shelby et al., 2014] Shelby, Z., Hartke, K., and Bormann, C. (2014). The Constrained Application Protocol (CoAP). RFC 7252.

[Standard, 1998] Standard, A. (1998). T1. 413,”. *Network to Customer Installation Interfaces-Asymmetric Digital Subscriber Lines (ADSL) Metallic Interface.*

[Sudevalayam and Kulkarni, 2011] Sudevalayam, S. and Kulkarni, P. (2011).

Energy harvesting sensor nodes: Survey and implications. *IEEE Communications Surveys & Tutorials*, 13(3):443–461.

[Suh et al., 2009] Suh, B., Won, C., and Kim, S.-W. (2009). Minimizing sleep duration time for energy harvesting wireless sensor networks. In *Sensors, 2009 IEEE*, pages 555–559. IEEE.

[Taleb and Kunz, 2012] Taleb, T. and Kunz, A. (2012). Machine type communications in 3gpp networks: potential, challenges, and solutions. *IEEE Communications Magazine*, 50(3):178 –184.

[Tan et al., 2011] Tan, H. O., Korpeoglu, I., and Stojmenovic, I. (2011). Computing localized power-efficient data aggregation trees for sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 22(3):489–500.

[Tan and Körpeoğlu, 2003] Tan, H. O. and Körpeoğlu, I. (2003). Power Efficient Data Gathering and Aggregation in Wireless Sensor Networks. *SIGMOD Rec.*, 32(4):66–71.

[Teklemariam et al., 2013] Teklemariam, G. K., Hoebeke, J., Moerman, I., and Demeester, P. (2013). Facilitating the creation of iot applications through conditional observations in coap. *EURASIP Journal on Wireless Communications and Networking*, 2013(1):177.

[Tsitsipis et al., 2011] Tsitsipis, D., Dima, S.-M., Kritikakou, A., Panagiotou, C., and Koubias, S. (2011). Data merge: A data aggregation technique for wireless sensor networks. In *Emerging Technologies & Factory Automation (ETFA), 2011 IEEE 16th Conference on*, pages 1–4. IEEE.

[Tsitsipis et al., 2012] Tsitsipis, D., Dima, S. M., Kritikakou, A., Panagiotou, C., and Koubias, S. (2012). Segmentation and reassembly data merge (sardam) technique for wireless sensor networks. In *Industrial Technology (ICIT), 2012 IEEE International Conference on*, pages 1014–1019. IEEE.

[Vasseur et al., 2012] Vasseur, J., Kim, M., Pister, K., Dejean, N., and Barthel, D. (2012). Routing metrics used for path calculation in low-power and lossy networks.

[Winter et al., 2012] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, J., and Alexander, R. (2012). RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550.

[Xia et al., 2017] Xia, N., Chen, H.-H., and Yang, C.-S. (2017). Radio resource management in machine-to-machine communications-a survey. *IEEE Communications Surveys & Tutorials.*

[Xiao et al., 2013] Xiao, M., Zhang, X., and Dong, Y. (2013). An effective routing protocol for energy harvesting wireless sensor networks. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 2080–2084. IEEE.

[Xu et al., 2009] Xu, X., Wang, S., Mao, X., Tang, S., Xu, P., and Li, M. (2009).