Network survival with energy harvesting : secure cooperation and device assisted networking

Filipe Conceicao

université PARIS-SACLAY

NNT: 2019SACLL020

Thèse de doctorat

Network survival with energy harvesting: secure cooperation and device assisted networking

Thèse de doctorat de l'Université Paris-Saclay
préparée à Télécom SudParis

École doctorale n° 580 "Sciences et Technologies de l'Information et de la Communication" (STIC)
Spécialité de doctorat: Réseaux, Information et Communications

Thèse présentée et soutenue à Evry, le 29 Novembre 2019, par

Filipe CONCEIÇÃO

Composition du jury:

M. L. MUÑOZ Professeur, Université de Cantabria (DICOM)	Rapporteur
M. P. LORENZ Professeur, Université de Haute Alsace	Président, Rapporteur
M. J. HOEBEKE Professeur agrégé , Université de Ghent (IMEC)	Examinateur
Mme. L. BOUKATHEM Maître de conférences, Université Paris 11	Examinateur
Mme. N. OUALHA Chercheuse, Commissariat à l' énergie atomique	Encadrant de thèse
M. D. ZEGHLACHE Professeur, Télécom SudParis	Directeur de thèse

Titre : La pérennité du réseau avec la récupération d'énergie : coopération sécurisée entre terminaux et mise en réseau sécurisée.

Mots clés : Securité, Energie, D2D, Récuperateurs d'énergie

Résumé : La technologie de réseau cellulaire de 5ème génération (5G) sera le réseau supportant l'Internet des objets (IoT). Elle a introduit une fonctionnalité majeure, communications appareil-à-appareil (D2D), que permettent communications sans fil à consommation d'énergie restreinte en interagissant à proximité et à puissance d'émission plus faible. La coopération entre appareils suscit donc un intérêt considérable pour l'énergie, et peut être utilisé en conjonction avec la récupération d'énergie pour prolonger la durée de vie des appareils. Les programmes de coopération renforcent la mise en réseau d'un appareil à l'autre, ce qui accroît la nécessité d'exécuter des mécanismes de sécurité pour assurer la protection des données et les relations de confiance entre les nœuds du réseau.

Ces mécanismes sont fondamentaux pour la protection contre les attaques malveillantes mais elles représentent aussi une importante consommation d'énergie, souvent négligée en raison de l'importance de la protection des données. L'établissement d'un canal securisé peut être coûteux en termes d'utilisation du CPU, la mémoire et la consommation d'énergie, surtout si les appareils sont limités en ressources. La confidentialité et l'intégrité des données ont un faible coût énergétique, mais sont utilisées en permanence. Il est donc nécessaire de quantifier la consommation d'énergie engendrée par la sécurité d'un appareil. Un modèle énergétique basé sur la sécurité est proposé pour répondre à cet objectif.

Dans les réseaux composés d'équipements d'utilisateurs (UE), la mobilité est une caractéristique clé. Elle peut agir sur la connexion à proximité d'objets IoT, étendant la couverture 5G vers l'IoT via les UEs. Une solution d'authentification légère est présentée qui permet par l'authentification directe et des communications UE-IoT, d'étendre la couverture et réaliser des économies d'énergie potentielles importantes. Cette approche peut être particulièrement utile en cas de catastrophe où l'infrastructure réseau peut ne pas être disponible.

La condentialité et l'authentification des données sont une source de consommation d'énergie importante. Les appareils équipés avec équipements de collecte d'énergie (EH) peuvent avoir un excédent ou un déficit d'énergie. La sécurité appliquée peut donc être ajustée en fonction de l'énergie disponible d'un appareil, en introduisant l'établissement de canal sécurisé qui tient compte de la consommation d'énergie. Après avoir étudié en profondeur les normes 5G, il a été constaté que les réseaux d'UE D2D utilisant ce type de norme dépenseraient une quantité

importante d'énergie et seraient généralement moins sûr. Un mécanisme léger de recléage est donc proposé pour réduire les coûts liés cette adaptation. Pour compléter le concept de canal sécurisé prenant en compte l'énergie et le mécanisme de recléage, une méthode de bootstrapping des paramètres de sécurité est également présentée. Le méthode désigne le cœur du réseau (CN) comme responsable de la politique de sécurité, rend l'ensemble du réseau plus sûr et aide à prévenir les pannes de communication.

L'adaptation susvisé requiert l'étude du compromis entre l'énergie et sécurité. À cette fin, un processus décisionnel de Markov (MDP) modélisant un canal de communication est présenté lorsqu'un agent choisit les éléments de sécurité à appliquer aux paquets transmis. Ce problème d'optimisation du contrôle stochastique est résolu par plusieurs algorithmes de programmation dynamique et d'apprentissage par le renforcement (RL). Les résultats montrent que l'adaptation susvisé peut prolonger de manière significative la durée de vie de l'équipement et de la batterie, et améliore la fiabilité des données tout en offrant des fonctions de sécurité. Une étude comparative est présentée pour les différents algorithmes RL. Puis une approche d'apprentissage Q-profond (DQL) est proposé que améliore la vitesse d'apprentissage de l'agent et la fiabilité des données.

Title : Network survival with energy harvesting: secure cooperation and device assisted networking

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France

Keywords : Security, Energy, D2D, Energy harvesting

Abstract : The 5th Generation Cellular Network Technology (5G) will be the network supporting the Internet of Things (IoT) and it introduced a major feature, Device-to-Device (D2D) communications. D2D allows energy-constrained wireless devices to save energy by interacting in proximity at a lower transmission power. Cooperation and device-assisted networking therefore raise signicant interest with respect to energy saving, and can be used in conjunction with energy harvesting to prolong the lifetime of battery-powered devices. However, cooperation schemes increase networking between devices, increasing the need for security mechanisms to be executed to assure data protection and trust relations between network nodes. This leads to the use of cryptographic primitives and security mechanisms with a much higher frequency.

Security mechanisms are fundamental for protection against malicious actions but they also represent an important source of energy consumption, often neglected due to the importance of data protection. Authentication procedures for secure channel establishment can be computationally and energetically expensive, especially if the devices are resource constrained. Security features such as condentiality and data authentication have a low energetic cost but are used constantly in a device engaged in data exchanges. It is therefore necessary to properly quantify the energy consumption due to security in a device. A security based energy model is proposed to achieve this goal.

In User Equipment (UE) D2D networks, mobility is a key characteristic. It can be explored for connecting directly in proximity with IoT objects. A lightweight authentication solution is presented that allows direct UE-IoT communications, extending coverage and potentially saving signicant energy amounts. This approach can be particularly useful in Public Protection and Disaster Relief (PPDR) scenarios where the network infrastructure may not be available.

Security features such as condentiality or data authentication are a significant source of consumption. Devices equipped with Energy Harvesting (EH) hardware can have a surplus or a deficit of energy. The applied security can therefore be adjusted to the available energy of a device, introducing an energy aware secure channel. After in depth analysis of 5G standards, it was found that D2D UE networks using this type of channel would spend a signicant amount of energy and be generally less secure. A lightweight rekeying mechanism is therefore proposed to reduce the security overhead of adapting security to energy. To complete the proposed rekeying mechanism, a security parameter bootstrapping method is also presented. The method denes the Core Network (CN) as the security policy maker, makes the overall network more secure and helps preventing communication outages.

Adapting security features to energy levels raises the need for the study of the energy/security tradeoff. To this goal, an Markov Decision Process (MDP) modeling a communication channel is presented where an agent chooses the security features to apply to transmitted packets. This stochastic control optimization problem is solved via several dynamic programming and Reinforcement Learning (RL) algorithms. Results show that adapting security features to the

available energy can signicantly prolong battery lifetime, improve data reliability while still providing security features. A comparative study is also presented for the different RL learning algorithms. Then a Deep Q-Learning (DQL) approach is presented and tested to improve the learning speed of the agent. Results confirm the faster learning speed. The approach is then tested under difficult EH hardware stability. Results show robust learning properties and excellent security decision making from the agent with a direct impact on data reliability. Finally, a memory footprint comparison is made to demonstrate the feasibility of the presented system even on resource constrained devices.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France

I acknowledge all researchers

Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

3GPP 3rd Generation Partnership Project

5G 5th Generation Cellular Network Technology

ANN Artificial Neural Networks

AES Advanced Encryption Standard

AES-CTR AES-Counter mode

AES-CBC AES-Cipher Block Chaining mode

AES-CCM AES-Counter with CBC-MIC

ADAM Adaptive Moment Estimation

APN Access Point Name

BAN Body Area Networks

BS Base Stations

BT Bluetooth

BLE Bluetooth Low Energy

CNT Core Network and Terminals

CN Core Network

D2D Device-to-Device

DL Deep Learning

DRL Deep Reinforcement Learning

DQN Deep Q-Network

DP Dynamic Programming

DQL Deep Q-Learning

EH Energy Harvesting

EM Electromagnetic

ECDH Elliptic-curve Diffie–Hellman

GW Gateway

IoT Internet of Things

IDS Intrusion Detection Systems

IPS Intrusion Prevention Systems

ICT Information and Communications Technology

MDP Markov Decision Process

MTC Machine Type Communications

MTCd Machine Type Communications device

MTCs Machine Type Communications server

MAC Medium Access Control

MIC Message Integrity Code

MSE Mean Squared Error

MLP Multilayer Perceptron

M2M Machine to Machine

MIKEY Multimedia Internet Keying

NIST National Institute of Standards and Technology

NFC Near Field Communications

OS Operating System

pmf probability mass function

PPDR Public Protection and Disaster Relief

ProSe Proximity Services

PKMF ProSe Key Management Function

PRR Packet Reception Rate

PSUE Public Safety UE

r.v. random variable

RL Reinforcement Learning

ReLU Rectifier Linear Unit

RAN Radio Access Networks

RF Radio Frequency

RFID Radio Frequency Identification

SP1 Smart Policy 1

SP2 Smart Policy 2

SA Services and Systems Aspects

SC Smart Cities

SHM Structural Health Monitoring

TD Temporal-Difference

tanh Hyperbolic Tangent

UE User Equipment

VI Value Iteration

WSN Wireless Sensor Networks

1 Introduction

This dissertation is part of the SCAVENGE project, a project supported by a Marie Skłodowska-Curie action. It deals with sustainable next generation of mobile networks, 5G. The main concern in the SCAVENGE project is energy and the use of renewable sources and in this dissertation, there is a focus on security issues. 5G networks will support Internet of Things (IoT) networks. Its deployment is envisioned in futuristic scenarios and international efforts have already been made to create worldwide standards to bring these networks to reality. There is a major concern with the current world wide use of environment detrimental energy sources and EH equipment constitute a viable, alternative choice to power all the electronic equipment that is required for mobile phones and IoT devices to work properly in the same ecosystem. However as it will be shown, significant security concerns are raised for these networks, along side with the energetic ones. This dissertation is about the study of these two vital aspects.

This section describes more in depth the project and the futuristic scenarios, explaining the importance of the advancements in the standards and giving an overview of the global energy production current situation and future forecast. It also provides the reader with an overview of the issues that will be discussed by describing a list of problems and contributions in this dissertation.

Sec. 2 provides a more in depth overview about all the aspects approached in this dissertation. The considered communication scenarios of 5G and IoT networks are described as well as their security related aspects. Special attention is given to MTC communication scenarios and their critical issues. MTC networks rely on low power radios and they are also addressed in the section along with their relevant security aspects. Finally, as a fundamental part of this dissertation, an overview of the state of the art of EH is presented. In this dissertation, several dynamic programming and machine learning methods are used to solve modeled problems. These methods require proper introduction and detailing. This is addressed in Sec. 3. Sec. 4 describes a state of the art survey that is connected to all the contributions presented in this work. The contributions are presented in the following four sections and Sec. 4 addresses all the work that was surveyed while developing these contributions. In Secs. 5, 6, 7 and 8, the referred contributions are then presented. Final conclusion remarks are made in Sec. 9 that include notes on all the contributions presented and Sec. 10 provides possible directions for future work, taking into account all the work developed in this thesis.

1.1 SCAVENGE

The SCAVENGE project was born from the idea of having a fully energetically sustainable 5G cellular network. Sustainable networks are based on the premise that environmental energy can be harvested through the use of dedicated EH hardware installed on the different network nodes that compose a 5G network and used to provide energy to power those nodes, namely Base Stations (BS) and the end devices. End devices are divided in this work in two main groups. They can be UEs or they can be any other network node that fits the category of Machine to Machine (M2M) Communications.

For the latter category, it is often seen in the literature the reference to MTC device or IoT device. In this dissertation, both terminologies are used and there should be no technological difference whether the term MTC or IoT device is used. IoT devices include sensors and all kinds of machines capable of wireless communications but they differ from UEs due to the latter's ability of making voice calls and being generally carried around in the physical environment by a person.

1.2 Smart Cities

The projected sustainable networks are envisioned to be in place in the years to come, applied to a broader spectrum of futuristic telecommunications scenarios such as Smart Cities (SC) and Public Protection and Disaster Relief (PPDR) scenarios. The SC paradigm is a way to manage the a city's infrastructures, its services and generally speaking, all its aspects. One key enabler technology for the concept of SC is the IoT because a significant number of different objects connected to the internet can be deployed virtually anywhere due to their small size and be used for sensing, actuating, computing or simply routing of information.

One of the main concerns of applications in SC is the limited amount of resources for the citizens, whether they relate to transportation, energy, food or any other commodity that is part of the daily life in a city. SC aim at optimizing these limited resources, providing effective and sustainable infrastructure and services while reducing natural energy resources consumption. It demands considering complex development of several areas such as governance, mobility, environment, people, economy and living in order to satisfy need of the city and of its citizens. A brief explanation of some of these concepts is necessary to understand the requirements in terms of communications.

Smart Governance is about using technology to facilitate and support better planning and decision making, empowering citizens to connect with government in new ways, through the use of social media, mobile apps, big data analytics and mash-up technologies [11].

The concept of Smart Mobility represents the improvement strategies for all citizen mobility aspects including the organization of public transportation systems, car sharing and use of alternative transportation means such as bicycles. To implement these strategies, the usage of applications to collect, store and process data, information and general knowledge is foreseen as the main instrument. The aim is to be able to plan, implement and evaluate integrated initiatives and policies of Smart Mobility.

Smart Environments are physical environments with devices capable of pervasive sensing, actuating and computing, and connected through networks for data collection, in order to enable various applications and services [12]. Typically considered Smart Environments are Smart Homes, Smart Offices, Smart Farms or Smart Hospitals amongst others. It is a sophisticated human-centric way of integrating hardware instrumentation with computational intelligence, for improved user-experience [13]. Smart Environment development requires wireless sensor networks to have cognitive capabilities and activity context awareness in order to efficiently optimize application performance and constrained resource usage of the sensor network [12].

There are also several examples of how the IoT can enable modern applications that can be of great usefulness to people, without being human-centric. One good example of this is Structural Health Monitoring (SHM), which is the process of detecting damage and location of the damage in engineering structures. SHM can be applied to monitor a great variety of structures, including many other than civil structures. For example, a bridge's structural health can be monitored in just the same way as an aircraft's structure.

All the described areas of smart development were described from a broad point of view. There is no need to detail further the scenarios and applications. It suffices to understand that SC make use of huge amounts of information to beneficially transform operations, work, and the life of citizens. The IoT represents an integrated smart system architecture of sensors, software, networks, and corresponding interfaces that holds the promise to be able to do just that. IoT systems can provide real-time awareness and integrate people, processes, and knowledge to enable collective intelligence for smart decision-making [14]. To be effective, smart systems need to be interconnected and intelligent, to enable the collection of timely high-quality data through embedded sensors that communicate over wireless or wired networks [14].

Still on the context of SC, a new paradigm for communications in emergency scenarios has been created. The concept of PPDR anticipates special communication needs for law enforcement, fire fighting, emergency medical staff and any other disaster recovery services, in the event of any emergency situation in public life, for these groups to better coordinate their operations in the disaster area. One key technology that has great potential for PPDR scenarios is Proximity Services (ProSe). ProSe consists of two main elements: 1) network assisted discovery of users who wish to communicate and that are in close physical proximity and 2) the communication between those users through direct communication (Device-to-Device (D2D)) and with or without supervision from the network. The data exchanged can also be voice data. The ProSe communication mode can be especially useful in remote areas not covered by terrestrial networks or when a backup transmission link is necessary such as in an event where the network is not available. Even when there is sattellite coverage, the high propagation delay makes call setup times longer and induces delayed data acknowledgement [15].

The role of the IoT in this context could be significant in providing information to build and maintaining the databases used by PPDR services, ranging from Environment monitoring, e.g. in floodings or earthquakes, Infrastructure monitoring with predictive alarms for maintenance in all kinds of civil structures, accidents localization and alarming, traffic information for optimization of routes in cases of emergency or video information from surveillance cameras. The IoT in PPDR also offers a feasible method to allow network connectivity in areas where terrestrial networks fail such as in underground caves or tunnels or any other difficult terrain so that during an incident there is a usable communications network where it would not otherwise be possible without existing emergency infrastructure in place. It can therefore provide enhanced network coverage in planned or unplanned events whereby terrestrial networks are unavailable or compromised.

Although its definition has changed during the past years [16–19], it has been shown that the interconnection of all the city related aspects are of great importance for the citizens of a SC [20]. Perhaps the most encompassing definition is that a SC is *a city in which the city dwellers may access smart services regardless of time or place* [21]. The role of communications is then obvious and Information and Communications Technology (ICT) becomes an essential part of the realization of that vision.

Naturally, there are some requirements for the connected IoT objects used in these applications that become obvious to be mandatory [22]. They are:

- EH capabilities for continuous available power;
- Long battery durability due to the difficulty of battery charging or replacement;
- Remotely manageable for security purposes and so that reconfigurations can occur;
- Sufficient memory and bandwidth for handling large amounts of sensing data;
- Resistant and very compact to be able to be mounted in small equipment;
- Open Source Operating System (OS) to allow customization for application;

This set of requirements is very difficult to suffice. As it will be explored, EH capabilities with small hardware, long battery durability and continuous connectivity are contradictory and strategies to respect all requirements need to be developed.

1.3 Communication Standards

The 3rd Generation Partnership Project (3GPP) is the body responsible for the development and management of the internationally used standards that define all technical aspects related to Radio Access Networks (RAN), Services and Systems Aspects (SA) and Core Network and Terminals (CNT). It becomes therefore essential to address these standards while designing solutions for future 5G networks.

When looking at these standards, one finds that the UE is usually regarded only as a 5G network node that communicates with the network infrastructure, the BS and Core Network (CN) elements, and with other UEs. The latter case is denoted as a ProSe network. This may however represent an under usage of its capabilities. Most UEs nowadays have numerous sensors. They can gather with those sensors a wealth of information about a user's context. For example, most modern smartphones and tablets can collect information related to geo-location, device orientation, mobility or light conditions. They are also equipped with several connectivity options such as Near Field Communications (NFC), Bluetooth, Wi-Fi and Cellular interfaces, giving them the ability to connect directly to a number of IoT devices. These capabilities and the fact they are carried by humans give them mobility attributes and make them excellent candidates to be a part of the IoT ecosystem, using D2D connections. Strategies can therefore be explored for D2D connections between UEs and IoT devices.

1.4 Global Energy Overview

The described futuristic scenarios are predicted to have billions of devices, all connected to the Internet and constantly generating low-rate monitoring, measurement, or automation data that many end-users and applications will frequently request [14]. The 5G technology will support 1000 times more capacity per unit area than the previous mobile networks generation, for more than 100 billion devices with typical user rates of 10 Gb/s, with requirements for significantly lower latency and higher reliability.

The Digital Power Group released a report in 2013 detailing trends in the ICT ecosystems and the world's energy generation, consumption and needs.

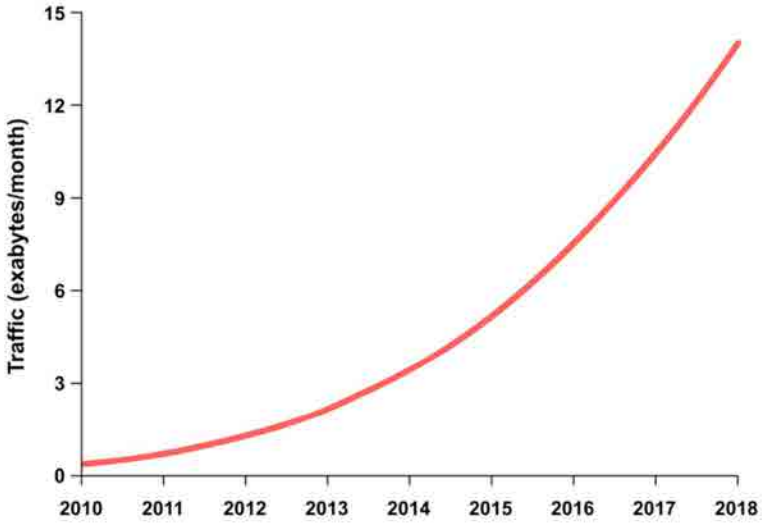


Figure 1: Global Mobile Data Traffic [1]

Fig. 1 shows the yearly global tendency in terms of mobile data traffic, portraying an exponential increase in world wide data traffic.

In Fig. 2, the monthly data needs are depicted taking into account the device type. It shows a clear trend of increased data requirements for UEs, tablets and machines, i.e., objects in the IoT. It is worth noting that the forecast for the IoT ecosystem is that it would have, in 2017, more than double of the data needs for smartphones in 2014. It is also worth noting that the data needs for UEs, tablets and machines have a much higher growth rate than that of laptops and desktop computers.

The higher capacity demanding human-centric communications will be achieved by the enormous expected growth in the number of IoT devices and access points. This will lead to an equally large growth in the carbon footprint of the ICT. The world's ICT ecosystem already consumes about 1000 TWh of electric energy annually, approaching

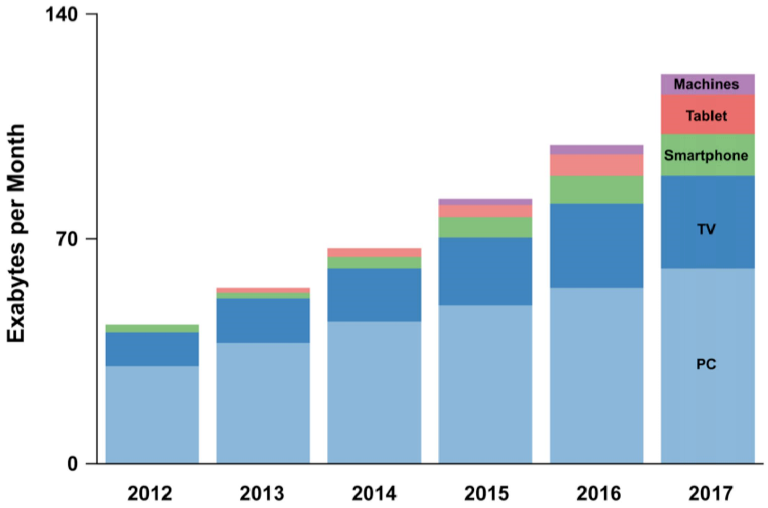


Figure 2: Yearly data needs per device type [1]

10% of the world electricity generation [1]. These numbers call for the urgent development of power saving and increased energy efficiency strategies to reduce the impact of ICT in the global energy consumption as well as to reduce its carbon footprint.

And because the carbon footprint is also a global concern, it is also worth analyzing the global electricity generation in terms of the energy sources used to produce that electricity. Fig. 3 shows different curves for the predictions for the global electricity production per energy source type. Coal and gas are two sources for electricity generation that when burned, are well known for being extremely polluting agents and sources of carbon into the atmosphere. It is foreseen that in the year 2035, these will still be the two main sources of energy worldwide. The presented figures showing increased tendencies call for solutions for the ICT using renewable sources of energy. In the case of the end devices, dedicated EH hardware installed on the devices is an alternative solution as an energy supply.

But it is not just the energy sources that are problematic and it is not enough to just use dedicated EH hardware to supply power to end devices. The general energy consumption from end devices is also shown to be a problem due to the contribution of ICT to the world's energy consumption. Moreover on the end device side, there are many devices which are considered to be resource constrained. The resource constraints do not only apply to memory and processing capabilities, but also to the low-power radio standards utilized that further constrain the network interfaces in terms of, amongst others, transmission distances. Therefore, on the end device side, the design of cooperation and device assisted networking schemes raise significant interest with respect to energy

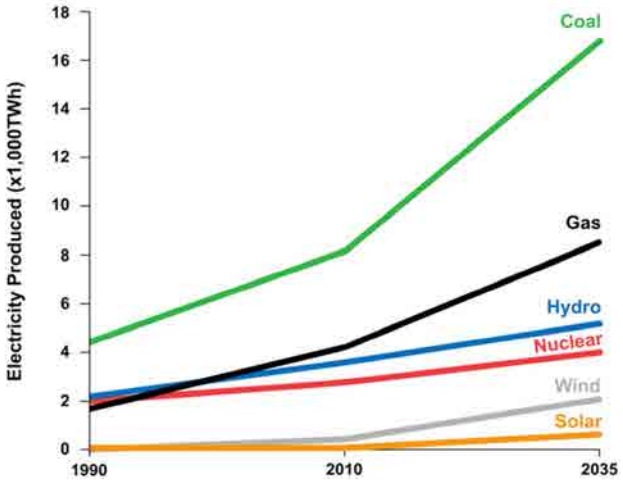


Figure 3: World's energy generation per energy type [1]

saving where D2D connections can be explored due to the fact that interactions take place in proximity having therefore a great deal of potential for that they communicate at a lower transmission power.

Despite the potential for savings, the exploration of cooperation schemes significantly increases networking between devices, raising security concerns. Trust relations between nodes need to be established and data protection needs to be assured. All nodes participating in cooperation schemes have security requirements that need to be fulfilled such as entity and data authentication, information confidentiality and integrity protection and these requirements need to be accounted for in all the interactions that the network devices participate in. The increase in usage of security mechanisms also has an obvious associated energy consumption. Security and energy are therefore the two main topics addressed in this work.

1.5 Problem Statement

This work addresses secure cooperation strategies between all end device types in 5G networks, exploring D2D connections as a means to reduce energy consumption while still providing secure communications. It also addresses the security-energy relation and trade off with the goal to achieve higher energy efficiency in this communication mode.

It is the understanding that both security and energy are important constituents

of telecommunications. The focus of security is to address and provide protection to systems, networks, programs and data from digital attacks. The predicted expansion of the number of communicating devices also expands the number of new systems to require this protection. Especially in the case of the IoT, where many of those devices are resources constrained, security solutions need to be investigated because the applicability of existing methods is not feasible or guaranteed due to their computational complexity and high energy consumption.

On the other hand, energy is a fundamental asset because without it, no network or device can operate. Along with the increase in the number of communication devices, data exchanged in 5G networks is also expected to grow by significant numbers, and also contributing to the previously mentioned expected increase in energy consumption from ICT worldwide. The application of EH hardware as local power sources in network nodes contrasts with the energetically sustainable 5G networks. Not due to its application itself but because the state of the art EH hardware that fits the requirements for end devices, especially in terms of size, is underdeveloped making these energy sources to be characterized as insufficient, erratic or intermittent.

The referred points composes the rationale for study of the security-energy relation for a good understanding of the impact of security on energy consumption, the study of secure cooperation strategies with the goal of reducing energy consumption as well as the study of the security-energy trade off to account for the cases where the available energy is not sufficient for normal device and network operation.

1.6 Contributions

This dissertation exposes and discusses the research motivation for the topics addressed on the relation between energy and security in D2D communications. The topics covered in this thesis cover a broad range of thematics connected to this relation and to the benefits attained from the cooperation strategies and the security-energy trade off. More precisely, the topics addressed are:

- D2D communications are regularly seen as UE-UE or IoT-IoT. In this work, UE-IoT connections are also addressed;
- the main security phases related to D2D communications, 1) security establishment and 2) secure communications, i.e., the data protection after a secure channel is established;
- the energy consumption of security primitives used for security establishment and secure communications;
- cooperation between devices in tasks such as routing data

The main contributions of this thesis relate to at least one of the mentioned covered topics. These contributions are:

- An energy model for IoT devices is introduced. The model enhances the state of the art as it is suitable for networking scenarios and it is based on security mechanisms, allowing for longer term conclusions on their impact on energy consumption of security establishment and secure communications phases and therefore enhancing the knowledge about the security-energy relation;
- A proposal for a lightweight security establishment protocol that can implicitly create secure D2D routes between an IoT device, a virtually unlimited number of UEs and the CN. The protocol introduces the idea of direct UE-IoT connections, exploring D2D to enhance aspects such as coverage increase, latency reduction and contributing to a generalized power saving potential;
- The introduction of the idea of making energy aware security decisions on the secure communications phase and the idea of eliminating energy consumption from some security features in that phase as an effective power saving strategy;
- A proposal for improvements directly in the 5G standards to allow for real time security context changes in the ProSe communications mode and that enables UEs to make energy aware security decisions in real time, as opposed to the fixed security policies enforced by the 5G standards for the secure communications phase;
- The application of a dynamic programming method to solve a communications model where, given limited available energy, energy aware security decisions are made to protect as much as possible transmitted packets in the secure communications phase while extending device battery durability and increasing data reliability as much as the available energy permits;
- The application of several Reinforcement Learning (RL) algorithms to the same communications model to study their applicability and performance in terms of battery durability, data reliability, provided security and learning speed;
- The application of Deep Reinforcement Learning (DRL) approaches to the same model to achieve faster learning, reduced memory requirements and more stability in the decision making during the learning process;

The modeling algorithms and tools used to achieve these contributions are mainly related to machine learning techniques. Their background and fundamentals are described in Sec. 3.

2 Background

The previous section depicted the futuristic scenarios where the sustainable 5G networks, powered by EH are envisioned to be deployed. The need for cooperation strategies and the security and energy concerns were also outlined. It is therefore very important to discuss important security aspects of end device communications and to discuss the state of the art of EH hardware. It is also imperative to discuss the state of the art related to the contributions in this thesis. They relate to energy consumption models and authentication protocols for the IoT, the 5G standards in relation to security policies and energy aware security decision making. A thorough investigation on the security aspects of 5G and low power radios was also carried via the study of the respective standards. All these aspects and tools are discussed in the following sections.

2.1 Security Overview

Security and energy in D2D wireless communications for end devices in 5G are the main topics of this dissertation. Therefore security is addressed in this work always in relation to its energy consumption. Obviously, security mechanisms are also addressed in terms of their functionality or purpose due to their role in offering data protection but the main goal of addressing security systems or mechanisms is in their relation to their energy consumption because of their regular usage in networking scenarios and the energy concerns also addressed. Hence, although cyber security can be addressed purely from the point of view of its functionality, the particular interest is in the security-energy relation in this work.

The key security concepts addressed also deeply relate with wireless networks and especially the type of networks addressed in this work such as IoT and ProSe networks. They are therefore related to key management, establishment and agreement, entity authentication and some security features that can be provided after a secure channel is established, namely confidentiality, integrity protection and data authentication. In this work, any different combination of these features in use while in communications, i.e., after security is established, is referred to as a security context.

Just like any other network functionality, the use of cyber security features or mechanisms will result in some related energy consumption. A thorough review of the available literature that evaluates energy consumption of security schemes or primitives reveals a common pattern of setting up testbeds with different communications devices, testing different primitives under the same conditions and recording the cost of the singular operations tested. Works presented in [7,23–29] are of this examples. Some of the values found in these works are presented in Tab. 1.

There are inherent problems with the quantification of energy consumed by security mechanisms found in the literature. The security mechanisms under evaluation mutate with time and newer versions of the mechanism keep being published and used in real systems. Then, new tests need to be performed to keep the literature up to date. Another difficulty relates to the platforms used for testing. It has been shown that the same mechanism being executed in different equipments will produce different results,

Table 1: Energetic cost of different AES modes of operation [7]

Key Size	Key Setup (μJ)	ECB ($\mu J/B$)	CBC ($\mu J/B$)	CFB ($\mu J/B$)	OFB ($\mu J/B$)
128 b	7.83	1.21	1.62	1.91	1.62
192 b	7.87	1.42	2.08	2.30	1.83
256 b	9.92	1.64	2.29	2.31	2.05

sometimes with a big gap between them [7,27]. Given the variety of end devices available and the low utility/effort ratio of performing experiments on all available equipments, the data found in the literature has to be considered as being merely indicative. Finally, although interesting to have this type of data available, it gives little information about the impact of security in a long term, after security mechanisms being executed several times due to the interaction with other network nodes.

2.2 5G Communication Scenarios and Security Related Architecture

In this section, the 5G communications scenarios are introduced and security aspects, including the security related architecture is discussed. The 5G security related architecture defines the key network elements that perform a role in securing the network. In this work, the particular interest is the security of end devices and therefore, in this section, this architecture is described from the functional point of view of providing security to D2D communications, and for both IoT and ProSe networks.

Fig. 4 depicts the considered end devices, UEs and MTC devices, and several options for direct D2D connections between them as well as the CN elements responsible for providing security for this communications mode, according to the 5G and MTC standards [3, 30]. The illustrated scenario shows the key elements in the CN and radio side. On the CN side, the MTC server is the element responsible for the security of MTC devices [30] and the ProSe Function is the element responsible for the general security rules and policies of ProSe communications [31]. The ProSe Key Management Function (PKMF) is responsible for the key management for ProSe communications [31] and is connected to the ProSe Function. For simplicity purposes, MTC devices and the MTC server are also sometimes denoted as MTCd and MTCs, respectively.

These end devices can be accessed from the radio network side through a BS. The BS provides coverage to a particular area within a certain radius. This area is considered to be in radio coverage. The area outside of this radius is considered to be outside radio coverage.

Then, it is important to describe the different communication modes considered in the 5G D2D standards and the ones presented in Fig. 4. All communication possibilities depicted in Fig. 4 are considered in this thesis. However, not all are considered in the 5G standards. In these standards, D2D communications are permitted between UEs only, and they can be between two or a group of UEs. Communications between a MTC device and a MTC server are also permitted. If a user needs to access data from one or

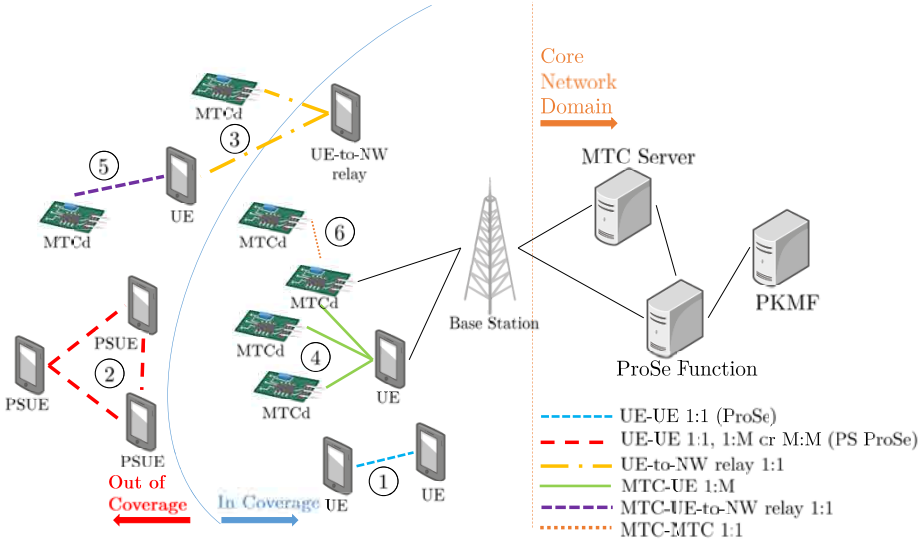


Figure 4: 5G Communication Scenarios and Security Related Architecture

several MTC devices, it has to communicate directly with the MTC server, via a BS.

Apart from the 5G standards, the scientific literature generally considers direct IoT-IoT communications. It can be therefore said that both the 5G standards and scientific literature consider two distinct groups, UEs and MTC devices and that D2D communications are of them exclusives. However and as described in Sec. 1, the mobility aspects and the radio interfaces present in UEs makes them excellent candidates to directly connect UEs to the IoT. Therefore direct UE-IoT are also considered in Fig. 4.

It is worth noting that D2D communications can happen in unicast, multicast and broadcast. In the 5G standards however they are referred to as 1:1, 1:M and M:M communications, respectively. These communications are addressed in this dissertation in the following way:

- Scenario 1) represents direct communications between two UE nodes and it is defined in 5G standards as ProSe. The contribution presented in Sec. 7 targets primarily this D2D connection and the main idea from contribution from Sec. 8 can also be applied;
- Scenario 2) is also defined in 5G standards and it is a network of two or more UEs that communicate directly outside of coverage from a BS. This scenario is envisioned to be useful in PPDR situations. The contribution presented in Sec. 7 also addresses 1:1 ProSe connections out of coverage and the main idea from contribution from Sec. 8 can equally be applied;
- Scenario 3) shows an UE inside radio coverage that is acting as a relay to other

nodes outside coverage. The coverage extension is defined in the 5G standards as a UE-to-Network relay but only for a UE-UE type connection. Sec. 6 proposes a security solution for allowing UE-MTC 1:1 type communications;

- Scenario 4) and 5) similarly to Scenario 3), in coverage and outside of coverage UE-MTC 1:1 type communications are addressed in Sec. 6;
- Scenario 6) is commonly found in the scientific literature but it is not addressed in the 5G or MTC standards [3,31]. This D2D connection is addressed in Sec. 8.

Scenarios 1), 2) and 3) (for the UE-UE case) are described in the 5G standards and their security is also well defined. In order to understand the contribution presented in Sec. 7, it is important to understand the security definitions from the standards for these scenarios.

2.3 ProSe security

Fig. 5 shows the signaling messages defined in the 5G standards to establish security between two UEs. The objective is to establish a root key, K_D, from which they derive further keys to use to protect their exchanged data.

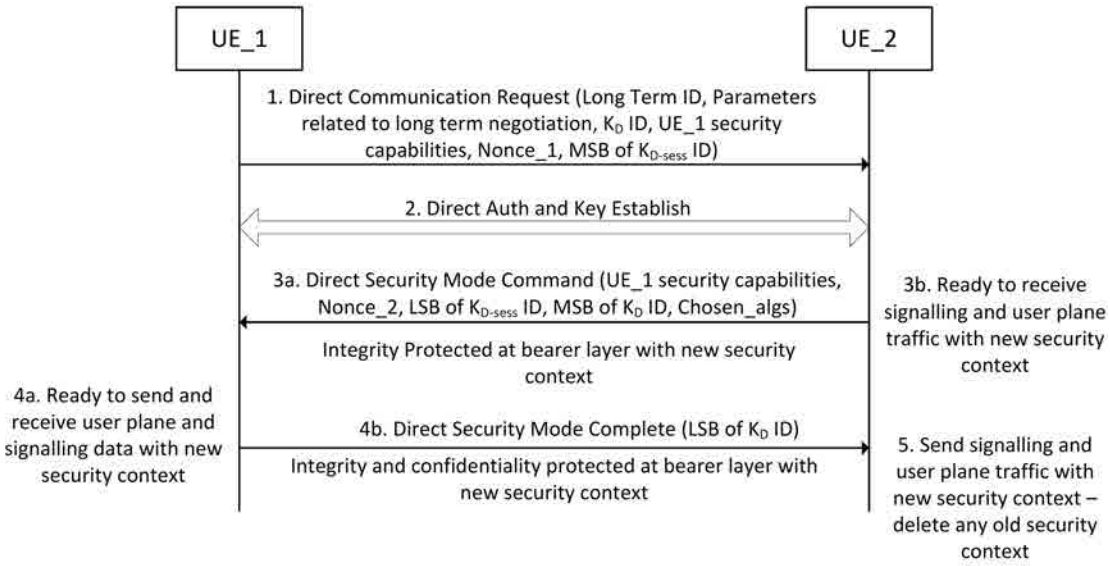


Figure 5: Signaling between UEs for K_D establishment [2]

In message 1, the UE1 security capabilities are a list of algorithms that UE1 will accept for this connection, to use in the Direct Authentication and Key establishment procedures, and the most significant 8-bits of the K_{D-sess} ID. These bits need be chosen so that UE1 will be able to locally identify a security context that is created by this procedure. After that, in message 3, the UE2 is sending the Direct Security Mode

Command to UE1. It includes the most significant bits of K_DID. If a fresh K_D is generated, $Nonce_2$ provides freshness to the session key being calculated and the $Chosen\_algs$ parameter indicates which security algorithms the UEs will use to protect the data. The included bits of K_DID have to uniquely identify the K_D at UE2. UE2 shall also return the UE1 security capabilities to provide protection against bidding down attacks. UE2 also includes the least significant 8-bits of K_{D-sess} in the messages. This bits have to be chosen so that UE2 will be able to locally identify a security context that is created by this procedure. UE2 calculates K_{D-sess} from K_D, $Nonce_1$ and $Nonce_2$, and then derives the confidentiality and integrity keys based on the chosen algorithms.

This procedure not only establishes security between the two UEs but also dictates the security context to be used in subsequent communications because the security context is always linked with a K_DID [31]. This means that, in order to change the security context, all these signaling messages need to be exchanged except for step 2, the Direct Authentication and Key establishment procedures, that are optional after the first time security is established [31]. This process is referred to as a rekeying process. This illustrates that UEs in ProSe are required to engage in heavy signaling if they are to change the security context that is currently in use. This signaling not only increases the energy consumption due to security overhead but it is also impractical to be executed very frequently, if the security contexts could be considered dynamic over a secure communications channel.

2.4 MTC communication scenarios

In the scenarios 5) and 6) of Fig. 4, MTC devices communicate directly. In the MTC standards however, communication where a user requires access to information from e.g., a sensor node, this communication is always done via an MTC server. This is depicted in Fig. 6.

In the case where there is no user involved and MTC devices communicate directly, this communication is done via the 5G network, as illustrated in Fig. 7. These communication modes can cause problems. Some relate to having a group of MTC devices providing information at the same time towards the CN, creating congestion issues. Others relate to mobility and the small batteries installed on the end devices that can create energy availability problems. Some of these problems are already predicted in the MTC standards and an overview is given in the next section.

2.5 MTC communications critical issues

In many MTC applications, a large number of MTC devices can be linked with a single MTC User, making the user affiliated with an MTC group. The MTC User associated with the MTC Group owns a MTC Server which is connected to the CN of a network operator via an Access Point Name (APN). The MTC devices in the MTC Group should be scattered over the network in such a way that the data simultaneously sent by the MTC devices in a particular cell is limited and will not cause a radio network overload. However, when a high number of MTC devices are sending or receiving data

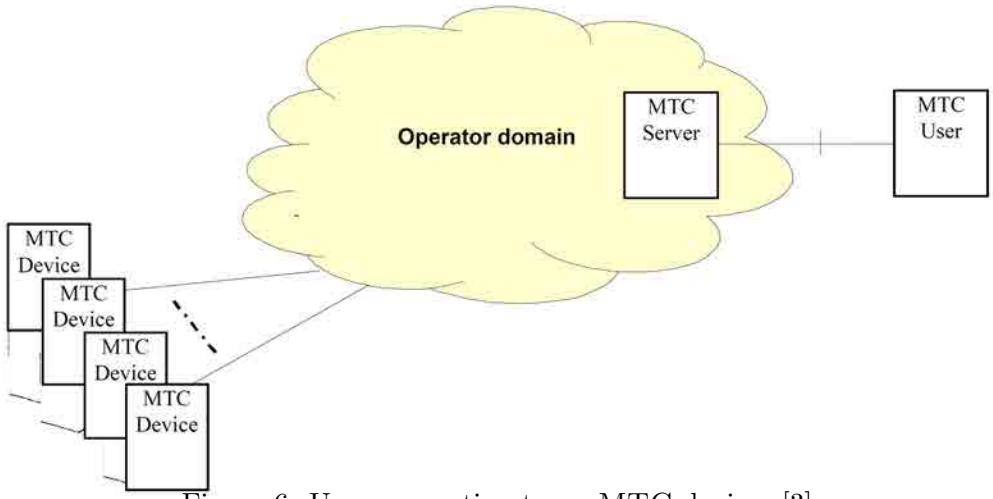


Figure 6: User connection to an MTC devices [3]

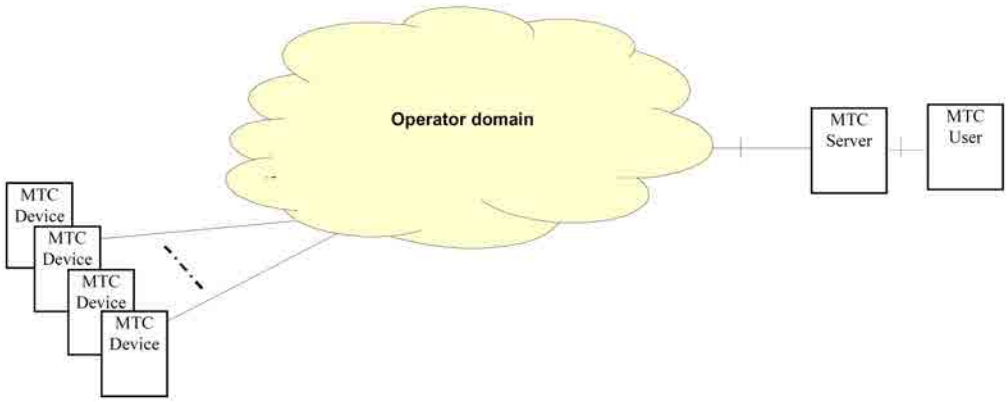


Figure 7: MTC devices communicating directly [3]

simultaneously, data congestion may occur at BS and CN levels, where the data traffic related to the MTC Group is aggregated [3].

In other applications, extra low power consumption is required. Several applications are already identified as problematic in terms of energy in [3]. Applications such as animal tracking require extra low power consumption because it is almost impossible to replace the battery or recharge the battery of a MTC device for animal tracking and using dedicated EH hardware is a possible solution. In applications such as cargo tracking, the cargo with a tracking MTC device could move very fast such as on a train and could stand still such as in a dock before loading or unloading. Extra low power consumption is also required as it is not desirable to either change its battery or replace battery during the transport period. Batteries of MTC devices for the tracking

Table 2: Low Power Radios Comparison [8]

	Range (m)	Throughput (kbps)	Power		Energy/bit	
			TX (mW)	RX (mW)	TX (nJ/bit)	RX (nJ/bit)
802.11G	30-100	54000	2300	1900	42.59	35.19
Zigbee	75	250	46.44	33.30	185.76	133.20
NFC/RFID	0.2	424	60.00	60.00	141.51	141.51
BT	30	2100	99.90	67.50	47.57	32.14
BLE	5	1000	48.00	39.20	48.00	38.20
Nordic RF	5	1000	21.47	25.65	10.74	12.83
BAN	5	1000	2.60	0.73	2.60	0.73

of elder people who have memory problem, children or pets could be charged or replaced. However, the worst case scenario is that they can go missing, requiring the MTC device to have a long working battery time in order to find them in case of disappearance. These are examples of high mobility applications.

For the low mobility case, extra low power consumption may be required for time controlled MTC devices. Time controlled MTC devices send or receive data only at certain pre-defined periods. MTC devices with this traffic pattern can be expected to receive non-periodic messages, e.g., emergency messages or notifications for altering the access periods. If the application requires the MTC device to send or receive data within pre-defined periods and receive non-periodic messages outside these periods, operation at the lowest possible power consumption level to extend battery life should be achieved.

Power saving strategies are presented in Sections 5, 6, 7 and 8 and the special case of mobility is also addressed more in detail in Sec. 5.

2.6 Low Power Radios

MTC devices will operate with low power radio technologies. The energetic cost of transmitting and receiving data has been reported to be the biggest source for energy consumption in IoT devices [32, 33]. Table 2 shows a comparison between different wireless standards currently developed and in use and that are aimed mostly at low power applications such as the IoT. The compared wireless standards are WiFi, 802.11 G, Bluetooth (BT), Bluetooth Low Energy (BLE), Zigbee, Radio Frequency Identification (RFID), Body Area Networks (BAN) and Nordic Radio Frequency (RF), a 2.4 GHz proprietary low powered radio.

The table shows that the numbers vary substantially for different standards. The 802.11G has transmission and reception power usage that is orders of magnitude higher than the other standards due to the fact that this standard was designed for high speed throughput and low latency, while the other standards in the table were designed for low power budgets [8]. Nevertheless, when comparing energy per bit, 802.11G has the same performance as BT. Zigbee has a rather high energy per bit requirement in order to

achieve its range of about 75 m. When comparing BT with the BLE version, the power consumption is much lower but at the expense of a lower throughput. This results in virtually no change in energy per bit.

2.6.1 Remarks on the Low Power Radios

Results displayed in Table 2 were obtained for a single platform, just changing the radio interface [8]. Changing the platform while studying the energy consumption of wireless low power radio interfaces can change the measured consumption significantly. The table demonstrates how difficult it is to obtain accurate energy consumption measurements that can be generalized for several different device types. It is therefore necessary to work and use values like the ones on Table 2 when quantifying energy consumption in different scenarios, as indicators of the cost per bit of the listed low power radios.

2.7 Security Contexts in Low Power Radios

The latest standards for low power radio technologies consider the possibility that the information being transmitted may not be fully protected all the time. For this, the concept of security levels has been introduced where different levels are defined that consider data protection via using or not different security features or mechanisms. In IEEE 802.15.4 based radios, this can happen after security is established. In the case of BLE, the security levels also consider the security establishment phase. In this section, these concepts are reviewed.

2.7.1 IEEE 802.15.4

The latest IEEE 802.15.4 based radio standard introduces an Auxiliary Security Header field on the Medium Access Control (MAC) frames with a variable length [9]. This field specifies the information required for security, including how the frame is protected, by means of a security level, and which keying material is used. There is also a Security Enabled subfield that, if enabled, will contain the information on how the payload is protected as defined by the security context selected for that frame [9]. Security contexts are used when devices operate in a secure mode. They cover confidentiality, integrity and data authentication and rely on AES security modes. Table 3 illustrates the valid security contexts with the Message Integrity Code (MIC) size in Bytes.

2.7.2 Bluetooth

In BT and BLE technologies, the rationale of not having to fully protect transmitted data is the same in BT and IEEE 802.15.4. However, security definitions differ in IEEE 802.15.4 and BLE, the considered standard in this work, presenting the concept of security mode, security level and pairing. Pairing is a process that can happen before communications take place between two devices and where all the security related aspects of that communication are defined. The pairing is considered in the definition of different security modes and levels in these standards [34].

Table 3: Security contexts defined in IEEE 802.15.4 [9]

Security Level	Security Suite	Confidentiality	Integrity	Data Authentication (MIC Size)
0	None	No	No	No
1	AES-CBC-MAC-32	No	Yes	Yes (MIC=4)
2	AES-CBC-MAC-64	No	Yes	Yes (MIC=8)
3	AES-CBC-MAC-128	No	Yes	Yes (MIC=16)
4	AES-CTR	Yes	No	No
5	AES-CCM-32	Yes	Yes	Yes (MIC=4)
6	AES-CCM-64	Yes	Yes	Yes (MIC=8)
7	AES-CCM-128	Yes	Yes	Yes (MIC=16)

There are two Security Modes, Security Mode 1 and Security Mode 2. There are also four security levels and each level can be associated with a Security Mode. Security Mode 1 represents data transmissions without signing the data. Security Mode 2 represents on the other hand the signing of the transmitted data, including both paired and unpaired communications.

The security levels are ordered from 1-4. Security Level 1 supports communication without security at all and communications are unpaired. In Security Level 2 communications are also unpaired but encryption is supported. Security Level 3 requires pairing and supports encryption. Security Level 4 also supports encryption and pairing, but with the mandatory use of Elliptic-curve Diffie–Hellman (ECDH) as the key agreement protocol [35].

2.7.3 Remarks on Security for Low Power Radios

The low power radio standards already foresee that some transmitted data may not require protection after security is established. Namely, the security features confidentiality, integrity protection and data authentication are emphasized. Providing protection from these features to packets adds extra information to the frames just before the physical layer reducing the available space for user (or device) data, increasing the security overhead and therefore, increasing the energy consumption due to security, after a secure link is established. The recognition of this aspect in the standards is also a motivation for looking at security from a different perspective and proposing that the use of security features can be reduced as an effective power saving strategy.

2.8 Harvesters

Although EH are a very promising technology for solving the energy constraints of traditional Wireless Sensor Networks (WSN), the power levels available from state-of-the-art energy harvesting devices is in the order of tens to thousands of μW or several

mW, corresponding to 1% to 20% of the required operating power which is not enough to power a sensor node continuously [36]. In order to fit the requirements that an energy harvester needs to have for its application on UEs or IoT devices, size is a very important consideration as it was shown in Section 1. End devices are small in size and therefore, there is a physical limit to the size of the EH hardware size that can be on them mounted. Especially in the case of IoT devices that tend to be smaller than UEs, size becomes a bigger consideration and being small becomes a bigger requirement. Fig. 8 shows a small harvester developed by University of Michigan. It was specifically designed to turn the cyclic motions of factory machines, i.e. vibrations, into electrical energy to power WSN.

Figure 8: A small vibrational EH [4]

Although size is not a direct correlation to the amount of electrical energy an EH can produce, being this small can give a very good sense that the energy production cannot be very powerful. Note that the coin in Fig. 8 has a diameter of $19.05mm$. It was found that there are three main types of energy harvesters that can be used on end devices based on fitting the size requirements and their research and development state. They are Thermoelectric, Electromagnetic Radiation and Vibration EH.

2.8.1 Thermoelectric

Thermoelectric technology converts the heat into electricity by the Seebeck effects, Peltier effect or Thomson effect [37]. Thermoelectric EH can be used to convert heat from electronic devices or human bodies or medical devices including e.g. hearing aids and cardiac pacemakers [38]. The power harvesting capabilities of these body-mounted devices range from $5\mu W$ to $1W$. The the main challenge of the thermoelectric power generation is the low heat-to-electricity conversion efficiency and a number of research efforts have been and continue to be undertaken to improve this aspect.

2.8.2 Vibration

Power generation from mechanical vibration usually uses ambient vibration around the harvesting device as an energy source and then converts it into electrical energy. The mechanical vibration either applies a force to a transducer or displaces an electromagnetic coil [36]. The harvesting method can be Piezoelectric, Electrostatic or Electromagnetic.

Vibrational EH are usually evaluated by their achievable power density. For an idea about the power that can be harvested with these harvesters, Tab. 4 shows a comparison between different EH types from [10].

Table 4: Power Density Comparison for Different EH Types [10]

Harvesting method	Power Density
Solar Cells (Outdoors)	100 mW/cm^3
Vibration (Piezoelectric/Electrostatic)	4μW/cm^3
Vibration (Electromagnetic)	800μW/cm^3

The inferior results shown in the table are mainly due to the fact that the resonant frequency of the generator is often not matched with the frequency of ambient vibrations or the frequency bandwidth of the generator is usually limited to a specific range which cannot, at times, cover the random ambient vibration's frequencies [39]. If the frequency of ambient vibration deviates slightly from the resonant frequency of the energy harvester, the resulting power output of harvesters is reduced drastically [40].

2.8.3 Electromagnetic Radiation

Electromagnetic (EM) harvesters have antennas that receive waves from RF radiation in the environment and then convert it to usable energy by means of rectifier circuits. The collected energy could originate from ambient radiation or from dedicated beam-forming signals emitted by a known transmitter. This kind of energy is available in reasonable quantities in urban environments, but can be scarce in sparse sub-urban environments. The harvestable power levels may be as low as $10^{-7}W$ for ambient sources, and the EM radiation is unpredictable and uncontrollable unless the emitting source and receiving antenna are static. It becomes therefore completely random and very difficult to model if the receiver is moving [41].

2.8.4 Remarks on the EH

From the studied EH sources, it is clear that the ambient energy can be very abundant but the harvesters are not yet mature enough to collect it in more significant numbers due to their low efficiency. It is still however possible to power an IoT device and work is ongoing to continue to improve the EH. Alongside with the efficiency, another concern is the instability of the amount of energy that is converted to electrical energy that the harvesters have, making the EH unpredictable as a power source. This aspect is especially linked with the Electromagnetic Radiation and Vibration EH types. Finally,

even if the EHs were more efficient, size is a limitation that does not allow them at times to provide enough energy for normal operation.

3 Modeling tools and Reinforcement Learning algorithms

In this section, the machine learning tools used in this dissertation are outlined. To achieve the contributions in this thesis, a thorough investigation and application of a dynamic programming method, Value Iteration, and several RL algorithms was made. These algorithms were used as tools to solve the problem described in Sec. 8 and are described in the following sections.

3.1 Markov Decision Processes

Markov Decision Process (MDP) are a formalization of sequential decision making where a system is modeled through a set of states, a set of state transition probabilities, an optional set of actions, a cost or a reward function and, for the case of infinite horizon problems, a discount factor parameter denoted by γ. Infinite horizon systems are systems that never reach a terminal state. If the model used has actions, the system evolves from state to state depending on the action selected and the corresponding state-action pair transition probability. After each transition, there is an immediate reward observed. MDP are an excellent tool to model RL problems because it allows that precise theoretical statements can be made [5]. The purpose of RL is to train an agent that 1) observes the way the environment progresses in time and the collected rewards, 2) chooses the appropriate action for each state of the environment and 3) learns in time how to make the best decision possible to maximize the cumulative rewards received. The agent–environment interaction is depicted in Fig. 9.

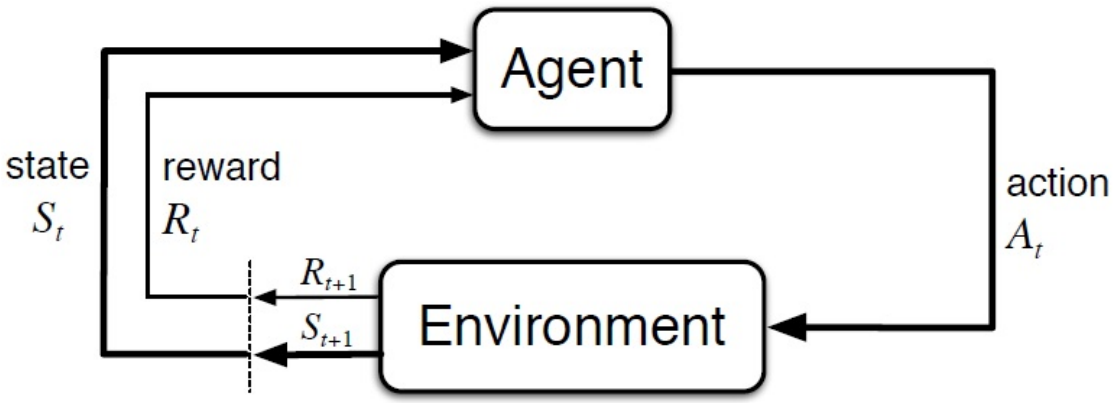


Figure 9: The agent–environment interaction in a Markov decision process [5]

The environment's progress is made in time slots and usually on episodic tasks, i.e, a finite sequence $S_0, A_0, R_1, S_1, A_1, R_2, S_2, \ldots$.

A state transition probability is defined as $p(s'|s,a) = p(s_{n+1} = s'|s_n = s, a_n)$, which corresponds to the probability of arriving at state s' knowing that the current state is s and action a_n is performed in slot n. The expected rewards for a given state-action pair are a function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$.

3.2 Learning - Returns, Policies and Goal

Once a system is modeled via a MDP, the goal is to make the agent learn an optimal policy that maximizes the cumulative rewards collected in an episode, i.e., the return. A policy is a mapping of an actions to every system state. If the policy is deterministic, after the learning is complete, the agent will always perform the same mapped action for any given state. The return is defined as the sum of the discount collected rewards during an episode as $G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$.

The action-value function is a mapping of real values to each state and the action-value function following a certain policy π is defined as the expected return collected following that policy, i.e., $q_\pi(s,a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a]$. The learned policy that maximizes the action-value function is termed the optimal policy, π_*, and it is the best possible choice of actions for each state that any agent can choose [5].

RL methods are algorithms for finding π_* from experience from interacting the environment, i.e. choosing actions, that are mainly used when part of that environment is not fully known. The unknown elements are usually the state transition probabilities. Hence, RL algorithms are coined online learning methods. But if there is full knowledge of the environment, the optimal policy can be attained by any Dynamic Programming (DP) method [5]. In the contributions of this thesis, Value Iteration was chosen as the Offline Learning algorithm.

3.3 Value Iteration for Offline Learning

Value Iteration updates the action-value function by iterating once over the state transition and reward matrices. DP methods are computationally expensive and can suffer from scalability problems [5]. Although there exists no mathematical proof to attest which algorithm is better for faster learning, Value Iteration is commonly regarded as the best option [5]. Pseudo code for Value Iteration will be given in Sec. 8.

3.4 Reinforcement Learning Algorithms for Online Learning

In the next sub sections, the RL algorithms applied in this work are briefly introduced. The used pseudo-code will also be depicted in Sec. 8. The reason for not exposing it in this section is that the generic pseudo code can be adapted to any problem with minor modifications.

These methods require a tradeoff between exploration and exploitation until convergence is reached. Exploitation means forcing the agent to tend to choose the action with the current highest immediate reward while in the learning process. Exploration on the other hand, means allowing the agent to choose actions that have lower immediate reward, so that it can arrive by itself to the conclusion of whether those actions are good options for maximizing the sum of the discount collected rewards.

The learning process has the necessity of using random action selection to ensure all state-action pairs are visited a theoretically infinite number of times to assure convergence [5]. A common way to provide exploration ability is by using an ϵ-greedy action

selection strategy. An exploration parameter $\epsilon \in [0, 1]$ is thus used and actions are selected with a probability $1 - \epsilon + \frac{\epsilon}{|\mathcal{A}_n(s)|}$ for the action with the highest value and a probability $\frac{\epsilon}{|\mathcal{A}_n(s)|}$ for all the others. The term $\mathcal{A}_n(s)$ denotes the number of possible actions for a given state in any given time slot.

A step parameter α_{TD} is also used to limit the weight of single state-action pair updates. This is commonly referred to as the learning rate. A discount-rate parameter γ_{TD} is also used to be plugged in the sum of the discount collected rewards formula, $G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$. Immediate rewards are denoted as R and will play a major role in the update of the $Q-value$ of a state. The $Q-value$ is a value used for an estimation of the action-value function during the learning process. In the following sections, that update rule is outlined. The notation used follows already the one used to describe the system model presented in Sec. 8 for easier consultation.

3.4.1 SARSA

SARSA is an on-policy method, i.e., it follows a policy and uses that policy to update the Q-values on every time slot based on the pair s_{n+1}, a_{n+1} where the action for the next state is chosen based on the policy being followed. The Q-value update is thus defined as:

$$Q(s_n, a_n) = Q(s_n, a_n) + \alpha_{TD}[R + \gamma Q(s_{n+1}, a_{n+1}) - Q(s_n, a_n)]$$

3.4.2 Expected SARSA

Expected SARSA is a variant from SARSA with a slightly different Q-value update rule. It takes into account the expected value of the action in state s_{n+1}. The expected value is calculated based on the action selection probability, $P(a|s_{n+1})$, that in this work comes from an ϵ-greedy approach. Follows that the update rule is given by:

$$Q(s_n, a_n) = Q(s_n, a_n) + \alpha_{TD}[R + \gamma \sum_a P(a|s_{n+1}) Q(s_{n+1}, a) - Q(s_n, a_n)]$$

By making updates based on the expected value, the variance of those updates is reduced and thus, in many cases, Expected SARSA tends to perform better achieving faster convergence.

3.4.3 n-step SARSA

The first version of online learning through experience was coined Monte Carlo method [5]. These methods make updates to the Q-values at the end of each episode based on knowledge stored during an agent's interaction with the environment. Keeping record of all the state transitions, actions taken and rewards collected results in a big increase in memory requirements. Furthermore, the same state-action pair can be visited multiple

times during the same episode which can easily result in a slow learning, due to the fact that the Q-value for that state-action pair is not immediately updated, which could result in choosing an under optimal action. For that reason, Temporal-Difference (TD) methods are widely regarded as faster learning methods [5]. In between these two ideas, n-step SARSA introduces a step parameter for evaluating Q-value updates n time slots in the future. In this way, updates are not calculated every time slot. Instead, the agent stores a small amount of information related to the experience, i.e., the states, actions and rewards observed during n time slots, and the update is calculated in future, delayed by the number of steps defined with n. This results in bigger memory requirements but it often shows faster learning results [5].

The Q-value update rule for n-step SARSA is given by:

$$G \leftarrow \sum_{i=\tau+1}^{min(\tau+n,N)} \gamma^{i-\tau-1} R_i$$
$$Q(s_\tau) \leftarrow Q(s_\tau) + \alpha_{TD}[G - Q(s_\tau)],$$

where G is the return and τ is the current time slot.

3.4.4 Q-learning

The Q-learning algorithm also accounts for the immediate reward and the current state-action pair. However, its updates differ in which they find the action tha maximizes the value of the next state, i.e. a greedier action. The Q-learning update rule is given by:

$$Q(s_n, a_n) = Q(s_n, a_n) + \alpha_{TD}[R + \gamma \max_{a_{n+1}} Q(s_{n+1}, a_{n+1}) - Q(s_n, a_n)]$$

Q-learning is one of the most widely used RL methods due to its simplicity and good results in different research areas.

3.4.5 Double Q-learning

Double Q-learning's principle is similar to that of Q-learning. However, this variant requires two action-value function, Q_1 and Q_2 and requires as well two Q tables to store its values. Their update rules are:

$$Q_1(s, a) = Q_1(s, a) + \alpha_{TD}[R + \gamma Q_2(s\prime, \arg\max_{a \in \mathcal{A}} Q_1(s\prime, a) - Q(s, a)]$$
$$Q_2(s, a) = Q_2(s, a) + \alpha_{TD}[R + \gamma Q_1(s\prime, \arg\max_{a \in \mathcal{A}} Q_2(s\prime, a) - Q(s, a)]$$

3.4.6 Remarks on RL methods

The presented RL methods are fundamentally used with the same purpose which is to attain an optimal policy. It is possible for certain problems that more than one optimal policy exists [5]. However, any of those policies can be used and the sum of the discount collected rewards obtained in an episode shall be the same. The main difference in the presented RL methods are in respect to their learning speed and as a consequence, in their performance. A method that learns slower will have a lower performance during the learning phase because it will make less good action choices. There is no theoretical proof in the scientific literature to show which one of these methods is more suitable to use on different problems. Therefore, a comparison between them is also useful in stochastic control optimization problems.

3.5 Deep Learning

The dynamic programming methods are proven to converge to optimal solutions in stochastic control problems and these methods are guaranteed to find an optimal policy in polynomial time [5]. Therefore, a dynamic programming method is exponentially faster than any direct search in policy space could be, because direct search would have to exhaustively examine each policy to provide the same guarantee. They can suffer however from memory related constraints due to the fact that the number of states often grows exponentially with the number of state variables, i.e. the characteristics of a state for a given system model, and due to the fact that many problems require a substantial state space in order to be a realistic model. The method presented in Sec. 3.3 requires that the transition probabilities and the reward or cost function are described in matrices and then operations are performed on those matrices to attain the optimal action for each state. As the number of states grow, the size of the matrices grows as well. As shown in Sec. 3.3, computational operations need to be executed. Several swipes over these matrices are required which can render these methods as either impractical or not suitable to simulate realistic systems.

On the case the RL methods, those computational swipes over the matrices are not required. In fact, these methods require only to have a table, the so called Q-table, that stores the current values for each state-action pair. As the learning is done online, only one entry of the matrices are updated at each learning step. This simplifies the computational effort while learning and reduces as well the memory requirements in comparison with dynamic programming methods. Despite the reduction of the requirements, there are IoT devices that can be severely resource constrained. Examples are Class 0 devices that can participate in Internet communications only with the help of larger devices acting as proxies, gateways, or servers [42].

DRL methods are a class of methods that further minimize the memory requirements for application of machine learning methods. That is because the Q-table is replaced by a function representation that requires less memory to be stored. The representation is the application of a function approximation technique that given an input, can give a good approximation of the corresponding Q-table value. There are different ways so

that the function approximation can be achieved. With linear methods, features are built so that the state space can be represented as a combination of state related values. Within non linear methods, Artificial Neural Networks (ANN) are often used and the latest advances in training deeply-layered ANNs are responsible for some of the most impressive abilities of machine learning systems [5].

An MLP is a a type of ANN that has interconnected units that have some of the properties of neurons, the main components of the human nervous systems. Hence, those units are called neurons as well. Figure 33 shows a generic feedforward ANN. It represents only one type of ANN that is used in this dissertation. There are however other types of networks that can be used for function approximation. There are no loops in the network, meaning that there are no paths within the network by which a neuron's output can influence its input. The network in the figure has an output layer consisting of seven output neurons, an input layer with two input neurons, and for which the number of layers of neurons in between can be variable. These are called hidden layers. The presented ANN is also fully connected, i.e., each neuron from the input layer has one connection to each of the neurons of the next layer, the first hidden layer. Then again each neuron of the first hidden layer has one connection to each neuron of the second hidden layer and so on, until the output layer is reached. A real-valued weight is associated with each of these links. The weights are a rough representation of the efficacy of a synaptic connection in a real human neural network [5].

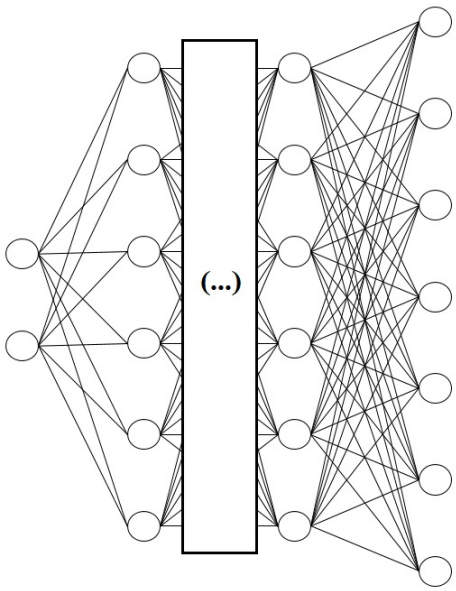


Figure 10: MLP model

The neurons compute a weighted sum of their input values and then apply to the result a nonlinear function, called the activation function, to produce the neuron's output, or activation. The functions used to calculate this results are therefore referred to as activation functions. There are several types of activation functions and there is

no theoretical support for the choice of function for better network performance while training. But typically, the Sigmoid, the Hyperbolic Tangent (TANH) and the Rectifier Linear Unit (ReLU) functions are among the most commonly used. The activation of each output neuron of a feedforward ANN is a nonlinear function of all the activation results over the connected neurons, starting from the input layer and moving towards the output layer, one layer at the time. It results therefore that the function that is being approximated is parameterized by the connection weights of the ANN and the weights are an effective way to represent the functions that are the object of the approximation via its features. Typically, the function being approximated is a state value or action-value function. In the latter case, it is denoted by $\hat{Q}(s, a, \mathbf{w}) \approx Q_\pi(s, a)$.

Training the hidden layers of an ANN is a means to adjust the network's weights. It is therefore a way to automatically create features appropriate for a given problem and therefore result in the desired function approximation. ANNs usually learn by stochastic gradient methods. With these methods, each weight is adjusted in a direction aimed at improving the ANN's overall performance as measured by an objective function to be either minimized or maximized. When used in RL, ANNs can use the TD errors to learn an action-value function. The TD error is used as the expected correct value in the output layer and the difference between that value and the actual obtained value from feeding an input to the input layer and calculating all the weighted sums in a feedforward logic until the output layer. Then in each training round, it is necessary to estimate how a change in each connection weight influences the ANN's overall performance, i.e., it is necessary to estimate the partial derivative of an objective function with respect to each weight, given the current values of all the ANN's weights. The gradient is a vector of these partial derivatives.

The process of then adjusting the weights is a backpropagation algorithm, which consists of alternating forward and backward passes through the network. In the forward pass, the weighted sums are calculated to achieve the current value of the output neurons given the current $\mathbf{w}$, i.e., $\hat{Q}(s, a, \mathbf{w})$. A backward pass computes a partial derivative for each weight. Each partial derivative is then used to adjust each of the corresponding weights until $\mathbf{w}$ stabilizes. At this point, the training stage is finished. As ANN can only approximate the action-value function, the policy attained at the end of the training is sub optimal. Nevertheless, results obtained by this methodology are often good enough considering the sub optimality and the attained benefits in terms of memory and learning speed [43].

The input layer is usually chosen to represent a state and the output layer is chosen, in case of control problems, to represent an action value. Given a state, the output values yield the value for each possible action. While learning or training is still ongoing, an action can be selected based on an exploration rule or a greedy one. Usually an exploration strategy is to choose an action randomly, but with the highest probability being to choose the action that yields the highest preference value. If the strategy is greedy, then the action selected is always the one that yields the highest preference value. In this case, the Deep Learning approach is Deep Q-Learning (DQL).

When dimensioning an ANN, some practical considerations are important. If the

ANN has a large number of weights, overfitting can happen. Overfitting is the problem of failing to generalize correctly to cases on which the network has not been trained [5] and to have more features than what the function actually requires [43]. If on the other hand the ANN has too few weights, then undefitting can occur. Underfitting is the problem of not having a minimum enough number of features that the function requires to be approximated. Moreover, the backpropagation algorithm does not work well in ANNs with a big number of hidden layers because the partial derivatives computed in the backward passes either decay or grow rapidly towards the input layer of the network, making the weight adjustment in the deep layers extremely slow or unstable, respectively [43].

There are no theoretical theorems quantifying how to choose the number of hidden layers and the number of neurons on those layers. It is therefore always necessary to perform some manual tuning of these parameters while testing a real code implemented solution.

4 Related Work

This section presents related work linked with all the contributions presented in Secs. 5, 6, 7 and 8. The contributions related respectively to Energy Models, Authentication Protocols, 5G standards for security policies and Energy Aware Security Decisions. The work reported in the next sections represents the state of the art investigated when the contributions were made.

4.0.1 Energy Models

One fundamental tool for understanding the energy consumption of an IoT device is an energy model. The available models in the literature focus on dividing a device into different blocks, usually hardware blocks, quantifying the energy of each block and summing it to obtain the device's total energy consumption. Examples of this approach are presented in [44], [45] and [46], where the models presented quantify the acquisition of data by means of transducers in the sensing block, the processing of that data in a processing block and the cost of sending and receiving information in the network in a communications block. The quantification of energy in these blocks is then summed up, quantifying the device's total energy consumption.

Quantifying energy consumption via hardware blocks disregards the basic characteristics of networking because no matter with how many and how much a device is interacting with other network elements, the energetic quantification is always the total for a given device. In other words, the calculated energy consumption is agnostic in relation to the network connections. This methodology is not helpful in the design of energy saving strategies when networking. Moreover, security is not considered in these models and therefore no conclusions can be extracted or made from the relation between energy and security.

A contribution on energy models is presented in Sec. 5.

4.0.2 Authentication Protocols

The coverage extension possibilities opened by the ProSe standard are not well explored. ProSe offers a way to offload some communications away from the BS. As for UE, the standard is clear now but MTC communications are still not well defined and have room for improvements. Although there are many solutions for resource constrained devices for key establishment in IoT environments [46], they do not involve the ProSe standard in 5G and hence, they do not involve the UE. The cooperation schemes for coverage extension in 5G are not abundant either. The works mentioned in this section all relate to 5G, establishing D2D security or simply authenticating Machine Type Communications device (MTCD) to the Core Network. However, they all differ from our solution presented in section 6 that provides security establishment for the UE-IoT direct communications case.

Authors of [47] propose a protocol for coverage extension where there are UE in coverage of a BS that serve as anchors to MTCD to send their data. A set of key

indexes are advertised by UE to devices outside coverage expecting that the receivers share at least one key with the sender. If there is no shared key, connection is not established. In this sense, this scheme is defined as a probabilistic key establishment scheme. Reference [47] proposes a cooperation scheme that allows for coverage extension based on a coalition of UE that cooperate and decide whether to accept or not to start a direct link connection with another device. This proposal relies on certificates and an asymmetric cryptosystem, generally considered computationally expensive for resource constrained MTCd. Work reported in [48] addresses authenticating MTCd towards Machine Type Communications server (MTCs) and provides mutual authentication between MTCd and MTCs. However, this solution does not consider the possibility of expanding coverage or direct communication with the MTCd. It needs 6 messages to authenticate devices with LTE radio capabilities and requires grouping the MTCd together by means of sharing a group key.

A contribution on authentication protocols is presented in Sec. 6.

4.0.3 Energy Aware Security Decisions

Authors of [49] propose a new research field coined *Green-Aware Security*. The purpose of the field would be both to evaluate actual security mechanisms considering their energy cost and effectiveness, and to build new security mechanisms that consider energy efficiency at their design stages. These approaches are discussed from a networking perspective in the broad sense, which therefore also apply to the IoT. The work points out that the modeling of security systems and mechanisms in terms of energy consumption is a largely unexplored field. The authors claim their work is a manifesto calling for amongst other things for 1) developing future generation security mechanisms optimised both in energy and efficacy and 2) defining new security solutions able to adapt their behavior based on security properties and energy consumption.

The work presented in this report is a perfect fit in this manifesto. In this section, we present related work that would also fit this manifesto idea. We intend to show however that these works address the topic of energy efficiency by improving existing mechanisms whereas we intend to go a step further by eliminating energy consumption from security mechanisms to zero under the assumption of no security threat present.

Work presented in [50] points out energy inefficiencies in key management schemes on ad-hoc networks. It shows that information on physical location of nodes can be used for energy-efficient key distribution schemes. The authors use a K-means approach to propose an energy-aware key distribution scheme and demonstrate higher energy efficiency. However, these findings are constrained to key management aspects which represent only a fraction of the overhead caused by security.

In [51], an energy aware authentication scheme is suggested to address energy constrained IoT devices. This approach is limited to security establishment.

In [52, 53], the authors propose an algorithm for route selection based on energy and security. In the first, authors define eight security levels, where the higher the level, the higher the energy consumption. The security levels are then combined with node residual energy for optimal path selection. Although this work addresses energy

efficiency, the security levels are defined to always provide security features, but reducing energy consumption by choice of security algorithms and key sizes. In the second, a multipath approach is proposed where two nodes alternate their communicating path based on security and energy metrics.

Works presented in [54–56] present energy models built based on security mechanisms. The first draws conclusions on the impact of security on D2D and routing path selection. The second proposes to create an objective function for a global power optimization problem, assuring secure communications. The latter proposes an actual optimization problem, with the goal to minimise energy consumed in D2D connections taking into account energy consumed by security mechanisms both due to computations and radio transmissions. These works are interesting references that call for energy efficiency increase and point the impact of security on energy consumption at the radio interface.

In [57], an energy-aware trust derivation scheme for WSN is proposed that uses game theory to minimize energy consumption in the network. This minimisation is however under the constraint of security assurance.

In [58], the authors present a trust and energy aware routing protocol using a distributed trust model for the detection and isolation of misbehaving and faulty nodes, and for route selection. The route selection is based on trust, nodes' residual-energy, and hop count, providing some results on reduced energy consumption and therefore increased network lifetime.

Work in [59] surveys energy efficient security mechanisms. All surveyed works on adaptive security rely on choosing energetically less expensive security mechanisms or protocols to adjust to the energetic status of a device or a network.

Contribution made on energy aware security are presented in Secs. 7 and 8.

5 A Security based Energy Model for the IoT

5.1 Introduction

The limitations and constraints of devices in the IoT and M2M networks are a major concern in research. Numerous articles can be found in the literature studying the topic and aiming at providing mechanisms for energy saving and increased energy efficiency. This results in less energy consumed by devices and by consequence, by their networks.

The referred limitations are usually connected to device's hardware. Memory, CPU clock, transceiver range and battery capacity are common limited features of devices in these networks. Hence, the effort in research to overcome these limitations is important. In the particular case of the battery, recent works show that for some applications, batteries can be replaced by energy harvesters and still maintain a device connected and operating properly throughout its life time. But in order for this to work, the device's energy consumption is a must know. This information can then be used for several purposes like battery or energy harvester dimensioning, task scheduling or for the any type of energy efficient strategies design. Works presented in [60] and [61] are examples of these strategies. In both, routing mechanisms are designed to achieve minimum power cost during data relaying from a source node to a destination. Both aim at maximizing the network lifetime by trying to keep a balanced distribution of residual energy of the network nodes. The needed balance can be understood in a 1-hop relaying for example. If there is one relay node that frequently or constantly offers to the source the best relaying cost, it will also be used constantly and its energy will be depleted faster, affecting the network life time dramatically [60]. This means however that these mechanisms do not always achieve maximum energy efficient as sometimes the best path (less costly) between source and destination is not chosen because it contains a node with its energy level under a certain minimum threshold. In the context of networks with energy harvesting capabilities, rejecting a path may not be the correct decision. Two major factors can alter the decision making: energy harvesting predictions and energy consumption due to networking tasks in several domains at the same time. The focus on this section is on the latter.

One fundamental tool for understanding an IoT device's energy consumption is an energy model. The models found follow a common behavioral pattern: acquiring data by means of transducers, data processing in a controller unit, sending and receiving information in the network and internal processes of the Operating System (OS) [62]. This leads to the quantification of energy in the identified consuming parts that, summed up, quantify the device's total energy consumption.

Although the energy consumption can be quantified in this way, it disregards the basic characteristic of networking. No matter with how many and how much a device is interacting with other network elements, the energetic quantification is always the total for a given device. This methodology is not helpful in the design of energy saving strategies when networking. Therefore, in this section, the idea of quantifying energy consumption based on its networking tasks is introduced. Acknowledging that a network node can have several connections in parallel with other network elements, quantifying

them separately is proposed. By summing each individual connection's contribution, the device's total energy consumption is quantified.

When starting a new connection, a procedure to establish a security context (e.g., session keys) is executed. This context is very important to be established as it can secure communications in different ways. An established security context is mandatory for authentication purposes for example, so that unauthorized access to the network (or free riders) can be prevented through entity or message authentication. For data integrity purposes which should be guaranteed not only against malicious alterations of data but also against passive threats originated by noisy channels that are subject to transmission errors and also for confidentiality, to guarantee only the intended or authorized parties can access information.

Obviously, these spend energy to be executed. And although the energetic cost of these operations may be relatively low in the case of symmetric cryptography, the cost of asymmetric techniques is not [7]. And especially after some networking time, when they are executed often enough, the amount of energy spent by them becomes significant. After a security context is established between two devices, other security related costs can occur. Message encryption and/or message authentication keep consuming energy during the life span of a device, while networking. Works presenting energy costs of the corresponding cryptography primitives show that they are not negligible [7], [29]. Yet, none of these aspects are considered in the existing energy models.

In the available radio technologies for proximity communications in IoT, Bluetooth Low Energy (BLE) and IEEE 802.15.4 based radios are at this point the dominating technologies. When evaluating the energy consumption of cryptographic algorithms, available works in the literature focus on quantifying single operations. Although these works are important, they are not enough to present a long term view of the implications that security has in networking and energy cost.

Due to these considerations, an energy model for IoT devices is proposed that slices the energy consumed by a device on each parallel connection it may have with other network nodes. The contributions presented in this section are as follows. 1) An energy model for IoT devices is presented that is suitable for networking scenarios, 2) The model slices the energy spent by each connection and maps it into the hardware blocks of the available models, 3) The model provides an energetic quantification method for the establishment, maintenance and termination of a connection with another network node, via the cost of cryptographic algorithms, 4) The model provides a quantification method for the cost of each connection while it is active, 5) The model provides a quantification method for security algorithms executed while a connection is active, 6) Some simulation results from networking scenarios in IoT are presented using the chosen cryptographic algorithms used in BLE.

5.2 Energy Model

The proposed energy model could be suitable for networking by considering all the possible interactions with all other network elements. Any node in a network can have several connections at the same time and this proposal is to quantify each one of them.

At a point in time, summing the energy consumption of all the n active connections, $E_C(x), x \in 1, 2, ...n$, with the energy spent by the OS in its routine tasks, E_{OS}, equals the total energy consumption of the device. This relation is given by Eq. 1.

$$E_{IoTd} = \sum_{i=1}^{n} E_C(n) + E_{OS} \tag{1}$$

Sensing and actuation, processing and networking are common energy consuming blocks in related works. A mathematical model to all, or to some of these blocks is presented in [45,63,64]. In this section, the same energy consuming blocks are considered but each active connection is mapped to them. Eq. 2a expresses this relation. Sensing and actuation, processing and networking blocks are represented as E_{SA}, E_P and E_{Net}.

Each connection between 2 devices comprises 3 phases. First, for security reasons, the establishment of security contexts for subsequent communication is performed. With the security context established, the connection is in the active phase and the devices now exchange data. While the connection is active, it can happen that the keys in use are renewed or the connection itself is no longer needed, and they are revoked. The energy consumed by connection establishment, maintenance and termination is denoted as E_{CEM}. The energy consumed in the secure, active phase is denoted as E_{SC}. Each connection added can increase the energy consumption due to application related tasks. This is denoted as E_{App}. Eq. 2b summarizes the energy consumed for each connection n.

$$E_C(x) = E_{SA}(x) + E_P(x) + E_{Net}(x) \tag{2a}$$
$$E_C(x) = E_{CEM}(x) + E_{SC}(x) + E_{App}(x) \tag{2b}$$

5.3 Mapping of the consuming blocks

The relation between Eqs. 2a and 2b is now addressed. Mapping the elements of these equations allows to connect the vision of a networking energy model with the hardware components of an IoT device described in the literature. In a device's architecture, sensing, processing and communication are commonly used energy consuming blocks [45,63,64]. In the proposed model, they are kept as the main consuming blocks but they are adapted to the introduced connection perspective.

Sensing and actuation tasks can be connection dependent and they affect only the sensing and actuation block. The total cost is the sum of the energy consumed by both tasks. and is given in Eq. 3.

$$E_{SA}(x) = E_S(x) + E_A(x) \tag{3}$$

In the Processing block, energy is consumed by the computations executed by active sessions. E_{CEM}, E_{SC}, E_{App} and E_{OS} are the energy consuming elements mapped to this block.

The communications block comprises the energy consumed by transceivers. The volume of data and radio interface can be connection dependent and may use different power levels. We therefore distinguish between energetic cost at transmission and reception, $E_{T\mathrm{x}}$ and $E_{R\mathrm{x}}$ respectively. Eq. 4 defines the cost of communications.

$$E_{Com}(x) = E_{T\mathrm{x}}(x) + E_{R\mathrm{x}}(x) \tag{4}$$

5.4 The role of security

As a fundamental part in all 3 phases of a connection, security plays a very important role in this model. The energy consumption of security algorithms is usually negligible when compared directly with radio communications for example. Nevertheless, it is important to quantify security costs because as time passes and networking interactions take place, the cost of security becomes not negligible anymore.

To account for the mentioned 3 stages of a connection, it is further defined that the connection establishment, maintenance and termination cost is the sum of asymmetric or symmetric connection establishment procedures, E_{ACE} and E_{SCE}, with the maintenance cost, E_{KM}. Termination cost is considered to be part of E_{KM} as it is usually a memory deletion operation with a very low energy cost. Procedures like public key verification are part of E_{ACE} to reflect the timing at which this procedure is usually done. Eq. 5 reflects this cost for each connection.

$$E_{CEM}(x) = \sum_{i=1}^{n} E_{ACE}(x) + E_{SCE}(x) + E_{KM}(x) \tag{5}$$

On the active phase of the connection, the volume of data encrypted and decrypted is denoted B_{Dec} and B_{Enc}. This data is encrypted and decrypted at the cost E_{Enc} and E_{Dec} per byte [7, 29]. Integrity protection and data source authentication, e.g., using a MIC or a digital signatures and their verification are also considered and denoted by E_{Int} and E_{DAuth} respectively. Entity authentication costs, if they exist in the active phase, are included in E_{KM}. Eq. 6 summarizes the cost of the active phase.

$$\begin{aligned} E_{SC}(x) =& \sum_{i=1}^{n} E_{Enc}(x).B_{Enc}(x) + \\ & E_{Dec}(x).B_{Dec}(x) + E_{Int}(x) + E_{DAuth}(x) \end{aligned} \tag{6}$$

5.5 Networking Simulations

In this section we present results from simulations obtained with well known software. We try to reproduce different cooperation scenarios in IoT networks that can be strategies for energy saving like in [60, 61] or to increase other network performance aspects like Packet Reception Rate (PRR). The devices are heterogeneous, i.e., both MTC devices and UEs, and in order to increase network performance and energy efficiency,

cooperation can happen between all devices, including direct MTC-UE, like introduced in [65]. The goal is to show that the weight of the security procedures on establishing and during a connection will impact in time the energy consumption of the overall network and should not be neglected. Results achieved are discussed in each subsection.

Fig. 11a shows different cooperation scenarios. All MTC devices are performing Periodic Updates (PU), i.e., MTC devices transmit updates to a GW on a regular basis with constant frequency and data size, as defined in [66] and illustrated in Fig. 11b. A frequency of 1 minute between data sending is used. Each minute is divided in 60 time slots ($1s$ each). The first $5s$ are used to transmit and receive information, as well as performing tasks like sensing, actuation or any computations needed. After $5s$ and until the next PU, the device's state changes to an Idle state. The daily energy consumption in the presented scenarios is then monitored.

The general IoT node OS' scheduling policy, each time slot is divided into smaller slots. Each of the smaller slots are reserved to the tasks that need to be executed concurrently, according to a given priority. Due to this policy, devices may appear to be multitasking but in reality, it is only true if we look at the completed tasks at the end of one complete time slot. For this reason, there is some freedom on the choice of time slot value. One second is chosen for simplicity but time slots can be changed to smaller values without changing the system's behavior or the achieved numerical results. The cost of a secure connection establishment, when it exists, is part of the cost of the first time slot for each PU.

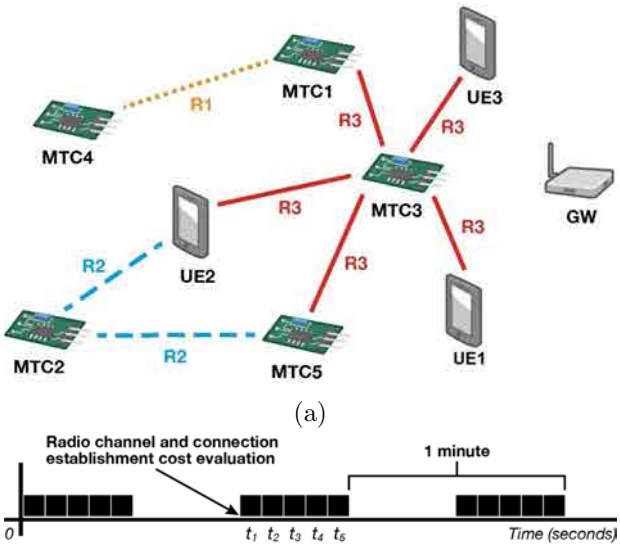


Figure 11: (a) General scenario system model.
(b) Periodic Updates representation.

In the simulations, $E_{App}(x) = E_{SA}(x) = E_{OS} = 0$. These values will vary between devices and although they would change the daily consumed energy, they are not relevant

Table 5: Simulations' parameters

Parameter	Value (Units)
B_{Tx}/B_{Rx}	$2(bytes/timeslot)$
E_{SCE}/E_{ACE}	$78.3 \times 10^{-7}/276.70 \times 10^{-3}(J)$
$E_{enc} = E_{dec}$	$1.21 \times 10^{-6}(J/Byte)$
$GW \rightarrow E_{Tx} = E_{Rx}$	$9 \times 10^{-6}(J/bit)$
$R_1 \rightarrow E_{Tx} = E_{Rx}$	$N \sim (2 \times 10^{-6}; 0.3 \times 10^{-7})(J/bit)$
$R_2 \rightarrow E_{Tx} = E_{Rx}$	$N \sim ([3, 12] \times 10^{-6}; 0.3 \times 10^{-6})(J/bit)$
$R_3 \rightarrow E_{Tx} = E_{Rx}$	$N \sim (2 \times 10^{-6}; 0.4 \times 10^{-6})(J/bit)$
$R_4 \rightarrow E_{Tx} = E_{Rx}(1\%)$	$N \sim (8.5 \times 10^{-6}; (0.028) \times 10^{-6})(J/bit)$
$R_4 \rightarrow E_{Tx} = E_{Rx}(5\%)$	$N \sim (8.5 \times 10^{-6}; (0.14) \times 10^{-6})(J/bit)$
$R_4 \rightarrow E_{Tx} = E_{Rx}(10\%)$	$N \sim (8.5 \times 10^{-6}; (0.28) \times 10^{-6})(J/bit)$

B_{Tx}/B_{Rx}- Data Tx/Rx in each time slot (bytes).
E_{SCE}/E_{ACE}- Cost of key establishment via symmetric/asymmetric algorithms (J).
E_{enc}/E_{dec}- Cost of encryption/decryption of data (J/Byte).
E_{Tx}/E_{Rx}Energy consumed to Tx/Rx data (J/bit).

for the security and networking interactions remarks we aim at showing. MTC devices establish a security context with the GW and use authenticated encryption and decryption of data in the active phase of a connection using Advanced Encryption Algorithm in Cipher-based Message Authentication Code (AES-CMAC). An AES key generation cost from [7] is used for context establishment. MTC devices establish connections with relays in the same manner and AES-CMAC is also used for ciphering. In case an asymmetric key agreement takes place, ECDH is considered. The cost of ECDH operations is taken from [67]. This choice of protocol and algorithm matches Security Mode 1 Level 4 of BLE security, that enforces an authenticated device pairing with authenticated encryption and is the recommended to be used by the National Institute of Standards and Technology (NIST) [34].

Table 5 lists the non zero parameters used in Eqs. 4, 5 and 6 in each of the scenarios presented in the following subsections.

5.6 Blind relay

MTC_1 in Fig. 11a performs PUs towards the GW. In addition, it relays data from a neighbor node, MTC_4, to the same GW every time it is requested, through link R_1. Channel conditions between devices and GW are assumed static and therefore, E_{Tx}/E_{Rx} are fixed. MTC_1 daily energy consumption is observed based on how often MTC_4 requests data relaying. This cost is compared with the cost of the PU, without relaying data for MTC_4, represented by the direct GW reference line. In Fig. 12 both energy consumption trends are plotted against the frequency of the relay requests.

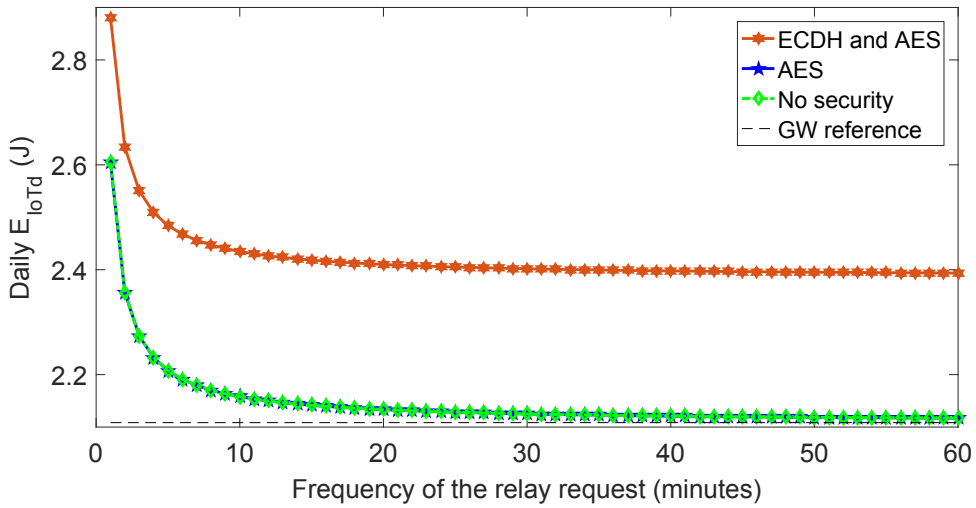


Figure 12: Energy consumption with and without relaying

The direct GW reference line is constant because it does not change with the frequency of the relaying requests. The cost of relaying data with no security context establishment, considering AES-CMAC only and both ECDH and AES-CMAC is also shown. As expected, the less often MTC_1 relays data, the less energy it consumes. But the plot shows different energy efficiency zones where 1 minute deviation in the frequency of the relay requests greatly impacts the energy consumed but after the 15 minutes frequency, the impact becomes smaller until it is almost negligible at 60 minutes. This idea can be extended to multiple MTC devices requesting relaying service at the same time, cause that the relaying load of MTC_1 oscillates. This may alter its decision on offering relaying service, causing it to advertise as a relay despite a low battery level. If the MTC device has energy harvesting capabilities then either load balancing can be improved using different energy efficiency frequencies or it might not be the best strategy to keep the distribution of residual node energy balanced. It can also help a node to understand when to be available to act as a relay.

5.7 The impact of security on a probable new relay

Despite the available strategies to routing information, relays are not always available to relay data. When they are, the probability of their availability changes for different applications, networking scenarios, energy levels, etc. In a smart city context, mobility of users is a characteristic of the networking scenario causing the probability of having relay UE available to change. In MTC networks with energy harvesting capabilities, MTC devices have at times full battery and active capability to harvest more energy and at others, low battery levels and no possibility to harvest energy. In this section, 3 different probabilities for MTC_2 to have a new relay available through R_2 are evaluated,

10%, 50% and 90%. MTC_2 will make the decision to relay the data block by evaluating the radio channel immediately before instant t_1, adding the cost of security due to a new secure connection establishment, comparing it with its GW link cost and choosing the lower cost.

The plot in Fig. 13a shows that even with a reduced probability for relay availability like 10%, energy savings of around 7% are achieved that may impact the life time of a constrained device. In all 3 cases, AES-CMAC connection establishment has a very small impact on the energy consumption. ECDH however has a noticeable impact. In this section only, the value of E_{ACE} (cost of ECDH) was 10 times the cost of E_{SCE}, instead of the one listed in table 5. Using the value in the table, E_{CEM} would be so high, that the device would never relay packets. MTC_2 would still decide to relay data establishing connections with ECDH key agreement with a cost of direct GW communications much higher due to a higher cost for the radio channel. But the parameters for the simulation were selected to fit resource constrained MTC devices and they show clearly that protocols like ECDH for key agreement can be energetically unaffordable for them.

With (90%) relay availability, the energy savings are up to roughly 60%. However, if load balancing techniques are applied like in [61], the energy savings will be partially lost because MTC_2 will at times refrain from relaying data. The plot also points out what is the difference in the radio link cost when using ECDH in the three cases, giving good insights on the budget for the use of asymmetric mechanisms.

Fig. 13b shows that in these conditions and decision method, the device benefits from using a value for R_2 slightly higher than the direct GW link, due to radio uncertainty in link R_2.

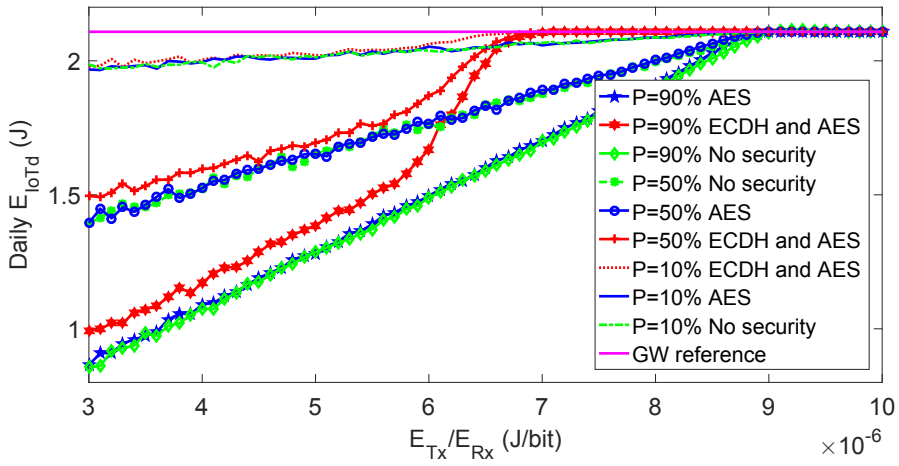
5.8 Concurrent transmissions

In this section, the energy consumed with PUs through the GW is seen as an energy budget that should not be exceeded. MTC_3 has 5 devices who offer better radio channel conditions (see table 5). The values used for R_3 are a simulation of a smart city scenario where often people (UEs) concentrate close to MTC devices and they leave after a while, e.g., public transportation stops. Instead of choosing the cheapest energy option, MTC_3 tries to maximize the number of relays to which it will send its data concurrently, without exceeding the energy budget.

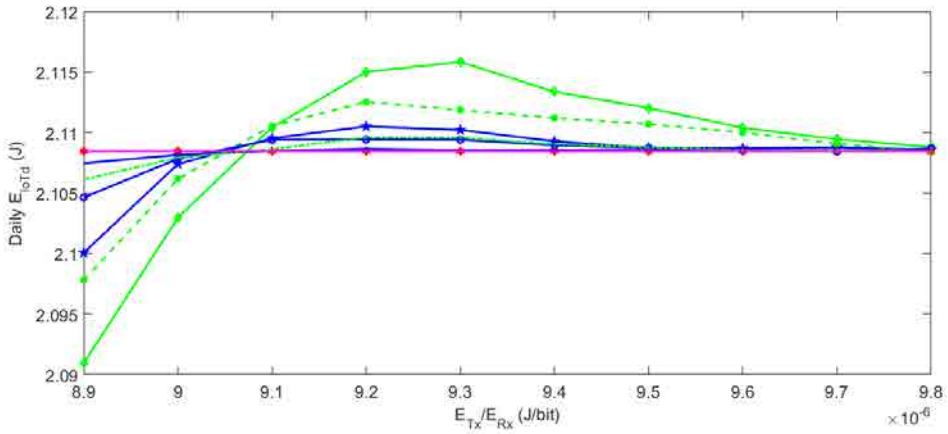
Concurrent transmissions of data in a wireless network reduces the negative impact of packet loss [68]. However, transmission of concurrent data can severely affect the lifetime of a MTC device. In case a node's battery is depleted, the lifetime of the other network elements is severely affected [69].

Immediately before t_1, MTC_3 checks and evaluates the radio channel of the available relays. Starting from the relay with the lowest cost and increasing, MTC_3 will transmit the PU to as many relays as possible, given that the cost of the transmissions plus E_{CEM} does not exceed the energy budget.

Fig. 14 plots histograms quantifying how many times one or more relays were used to transmit data daily. If more than one was used, the remaining carried redundant



(a) Energy savings for different relaying probabilities



(b) Relaying decision limit

Figure 13: Energy savings due to relaying

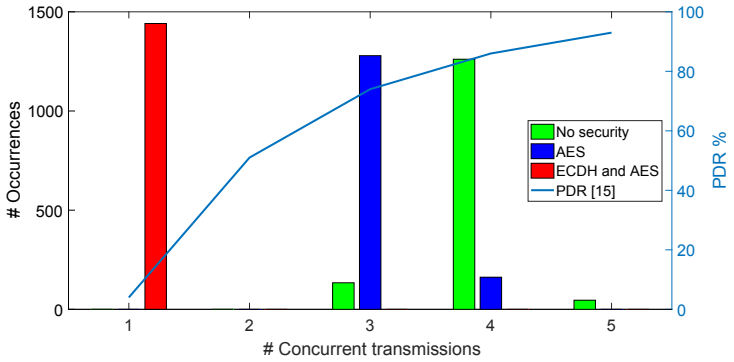
data. Figs. 14a, 14b and 14c plot histograms for 2, 10 and 20 bytes of data sent.

Work presented in [70] shows simulation results with the PRR as a function of the number of concurrent transmissions. This result is used in the plots to link the PRR to the number of relays used by MTC_3.

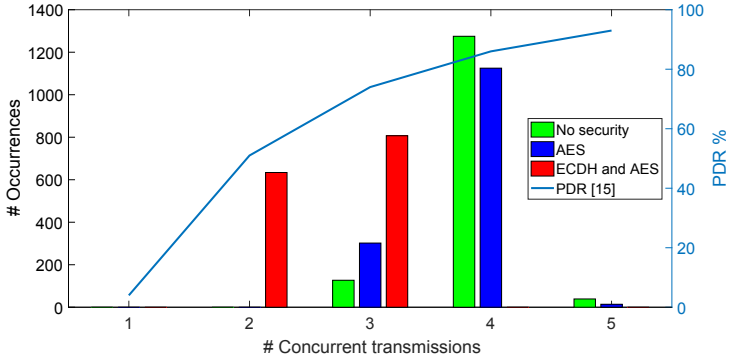
When only an AES-CMAC key establishment is considered for 2 bytes per time slot, the maximum number of relays used by MTC_3 varies between 3 and 5. However, if ECDH key agreement takes place as well, only 1 relay is used, showing that the weight of an ECDH key agreement has a clear impact on the effort of increasing PRR. However, as the amount of transmitted data increases, increasing E_{SC}, the concurrent transmissions increase as well. This means that E_{CEM} has a different impact on the PRR depending on the value of E_{SC}. Therefore, relations can be made between the cost of the active phase and the establishment of a connection. PRR can be improved based on reducing E_{CEM}, increasing E_{SC} or this can be used to calculate the energy budget for security in some IoT applications.

5.9 Remarks on the Energy Model

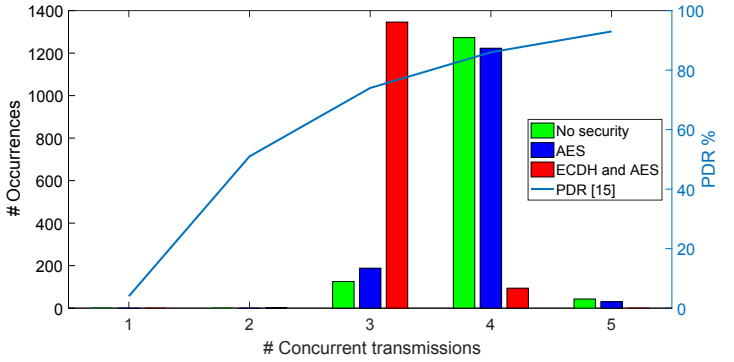
A new energy model for IoT devices has been introduced in this chapter. The model serves as a tool to quantify the energy cost of establishing connections and of their active phase. It quantifies as well the cost of all modern cryptography algorithms executed for all connections. The model allowed to conclude aspects related to relaying and concurrent transmissions decisions. Techniques like load balancing or constraints like minimum thresholds for battery level are shown not to be the best strategy all the time. Presented results consider BLE security protocols and algorithms, demonstrating that it is not affordable for all devices.



(a) 2 bytes Tx per time slot



(b) 10 bytes Tx per time slot



(c) 20 bytes Tx per time slot

Figure 14: Histograms - Daily number of occurrences for different concurrent transmissions

6 Security Establishment for IoT Environments in 5G

6.1 Introduction

In [71], a projection related to 5G and the IoT predicts 29 billion connected devices by 2022. Fom this number, 18 billion will be related to the IoT. Connected devices include cars, machines, meters, sensors, point-of-sales terminals, consumer electronics and wearables. By the same year, a worldwide total of 6.2 billion (all different) mobile subscribers will hold a total of 9 billion subscriptions. With these predictions for IoT devices and subscribers, the connections between devices is expected to increase within IoT networks or in interaction with other types of equipment.

For direct connections between UEs, 3GPP has standardized D2D communications naming them ProSe [2]. For MTC, it has released the recommendations for security mechanisms [3]. In the MTC category, the architectural model consists of a client, the MTC device (MTCd), and a MTC Server (MTCs) that is responsible for the security of a group of MTCd. The MTCs can also store particular information sent from each MTCd under its control [3]. This operational mode doesn't account for volatile data or actions that don't need to be recorded and increases, in a general way, the latency. It compels a user who needs information from a group of MTCd or wants to interact with them, to run a security procedure with MTCs to get needed data, through a BS. MTCd are expected to authenticate to a MTCs and send their data or receive commands from it. In parallel, a user carrying a UE and authorized to interact with certain MTCd, also needs to authenticate to MTCs. After this procedure is complete, the interaction between MTCd and UEs runs through the MTCs, rather than directly, in a D2D fashion. The MTCs is a participant in the user plane (UP) data flow, which adds energy and bandwidth consumption. As an example, using the simplified path-loss model [72] with a reference distance $d_0 = 10m$, constant $L = 4.38 \times 10^{-8}$ and $\gamma = 3$, and comparing losses for communication distances, e.g., $20m$ for D2D and $300m$ for a cellular link, we see that a D2D link has a loss roughly 3425 times inferior than a cellular link.

In the 3GPP architecture the MTC UP data is required to flow through a server beyond the BS to reach back to an UE. There is always a direct connection between the MTCd or a GW to a BS, without room for cooperation schemes that could allow for reduced power transmissions and bigger coverage area. If we equate mobility of the MTCd, as in any moving vehicles, devices installed in moving parts or even wearables, we see that MTCd communicating directly with a BS can have a very high cost in terms of power. Some devices will simply suffer from power depletion. However, if they could directly connect to another device for their routine interactions, the power saved could be significant.

Therefore we see an opportunity to shorten communication distance, using the potential of the ProSe functionality. We look at the numbers estimated for the IoT, mobile subscriptions and scenarios in smart cities and PPDR use cases and foresee a bigger number of interactions UE-MTCd, many times higher than the number of deployed devices. These connections need to be secure, even if just to guarantee the integrity of the messages exchanged. Therefore the number of end-to-end pairwise keys is, regardless of

the technique used, bigger than the number of users interacting with other devices.

Therefore, in this section, the focus is on the key establishment for connections between UEs and MTCd aiming at reducing the communication costs of these connections by taking advantage of proximity and the ProSe standard (that cannot be used with resource constrained devices), and a lightweight key distribution scheme is presented. Service authorization and authentication of all participating devices is accounted for, i.e. MTCd, UEs or GWs. Specifically, the proposal in this section is of a protocol for mutual authentication of a MTCd and a MTCs using an UE as a relay. At the same time, the MTCd and the UE establish a symmetric master key. This allows them to communicate directly, making it possible for the MTCs not to participate in the UP data flow. A cooperation scheme is also presented that is based on the proposed protocol that extends 5G coverage towards MTCd. The main contributions of this section are: 1) a method to distribute a shared secret between each MTCd and an UE that wishes to communicate with them is provided. The key pair is symmetric, respecting the resource constrained nature of MTCd. 2) The method provides authentication and authorization services of all participating devices. 3) The proposed solution is able to resist to known attacks. The automatic protocol formal verification tool ProVerif was used to prove our protocol's security. 4) The presented solution limits the communication range of MTCd or a GW to an UE in proximity, rather than a BS. It also removes the MTCs from the UP data flow to save energy in the MTCd, the UEs and in the overall 5G network.

6.2 System model

In the envisioned system model, UE, MTCd, GW and MTCs coexist in radio range and can initiate communications with all surrounding nodes as depicted in Fig. 15. The considered network elements can be part of the 3GPP network or not [3]. This is an important consideration, e.g., for the roaming cases where any of the elements may belong to another network and still be allowed to connect directly.

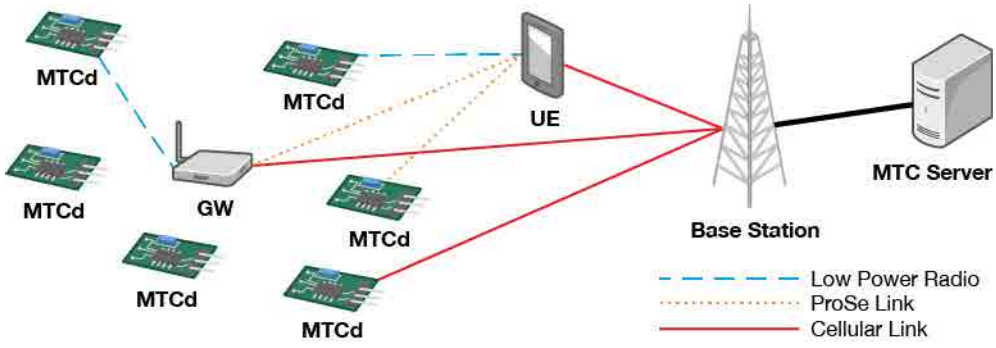


Figure 15: System Model

The UE is considered to be any device that has, amongst others, 5G and LTE radio interfaces and is usually held and controlled by a person. The MTCd are devices with low power radio technologies like the ones based on IEEE 802.15.4, Wi-Fi, Bluetooth,

etc., and may have 5G and LTE radio interfaces. Regardless of their embedded radio interfaces, they are considered here as to be resource constrained in terms of computational capabilities, memory space and battery capacity and to be equipped with Machine Type Communication capabilities [3]. GWs are devices that can serve as a radio GW for a cluster of MTCd in their vicinity. The GW role is to receive information from MTCd, and send the information through a more powerful, longer range LTE or 5G radio channel, to a BS. The MTCs is the element that is assigned a number of MTCd, and that is responsible for the security of these devices and/or storing their data.

6.3 Authentication Protocol

The proposal for a protocol to directly connect UEs and MTCd is now detailed, including its messages and their content, as well security features of the proposed solution. The notation used is shown in table 6. The protocol has both an initialization and key exchange phases and it is used for an IoT coverage extension, using the ProSe standard.

Table 6: Notation used

Abbreviation	Definition
PSKey	Pre shared key
DMKey	Derived Master key
MIC	Message Integrity Code
MTCdMIC	MIC calculated by MTC device
MTCsMIC	MIC calculated by MTC Server
MTCdID	MTC device's ID
MTCsID	MTC server's ID
UEID	UE ID
GWID	GW ID
MTCdNonce	Nonce generated by MTC device
MTCsNonce	Nonce generated by MTC server
KDF	Key Derivation Function
MTCdInfo	Information used by the device
MTCsInfo	Information used by the server
||	Concatenation

6.4 Initialization phase

The initialization phase consists of a set of pre-determined conditions representing the assumptions made during the design of the protocol and are mandatory for it to run properly. Namely, the following is considered:

- Each MTCd has at least one MTCs responsible for security material distribution, authentication and authorization of the MTCd assigned to it [3];

- Each MTCd has a pre-shared secret key that is only known to itself and to the MTCs it is assigned to;
- Each MTCd is assigned with an unique ID inside its own MTCs cluster and knows its MTCs IDs [3];
- The UEs have access through a secure channel to the MTCs under the 3GPP system responsibility [3];
- In UE-UE or UE-GW direct links, the communications use a secure channel established according to ProSe [73] [3].

It is also assumed that an MTCd can start communication with another device and indicate that its radio channel evaluation is out of the scope of this work. It is important to note that all the assumptions made are part of the 3GPP MTC standards [3], exception being the pre-shared key assumption, that is in any way a reasonable and widely used assumption in the design of security protocols.

6.5 Key exchange phase

In the key exchange phase, the protocol is simply composed of 4 messages. These messages are represented in Fig. 16 and a description of their content and purpose follows.

Message 1: MTCd generates MTCdNonce (for freshness of the message) and uses it with PSKey, MTCdInfo and the IDs of the participants in the protocol to calculate MTCdMIC keyed with PSKey. In this way, the IDs of all participants in the routing path are binded to mitigate the risk of spoofing attacks.

$$\textbf{MTCdMIC} =$$

$$MIC(PSKey, MTCdID, MTCsID, UEID, MTCdNonce, MTCdInfo)$$

MTCdInfo may contain information about the UE/GW ID connecting to MTCd, contextual information (e.g., location) or any other that the MTCd needs to send to MTCs, as for example related to the cryptographic algorithms supported (e.g., MIC algorithm, KDF function) or its current status (e.g., battery level). No particular algorithm for MTCdMIC calculation is particularly advocated although the the recommended algorithms in [74, 75] are strongly suggested. MTCd then computes and sends to the UE:

$$\textbf{M1} = MTCdID||MTCsID||MTCdNonce||UEID||[MTCdInfo]||MTCdMIC$$

Message 2: When the UE receives **M1**, it simply forwards it to the MTCs with identity MTCsID.

Message 3: Upon reception of **M2**, MTCs can check the UE's service authorization with the Core Network. Therefore, mutual authentication between MTCs and UE can

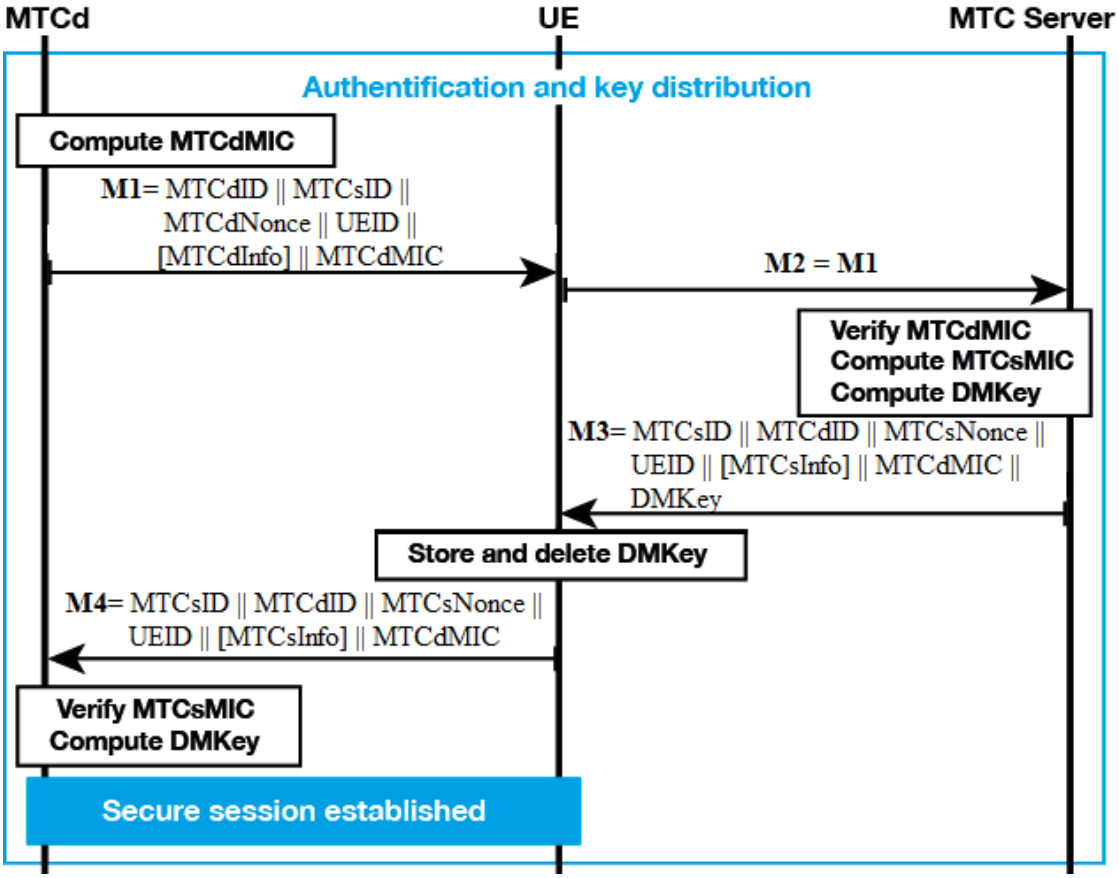


Figure 16: Authentication protocol

take place (e.g., using TLS/SSL). Then, it verifies MTCdMIC using the PSKey and the elements in **M2**. If the challenge was correctly answered by the MTCd, its authentication in MTCs is completed. It then generates MTCsNonce and calculates its own MIC:

$$MTCsMIC = MIC(MTCdID, MTCsID, MTCdNonce, MTCsNonce,$$

$$UEID, MTCdInfo, MTCsInfo, PSKey)$$

MTCsNonce and IDs are used again for the novelty of M3 and to bind IDs. MTCs then generates DMKey:

$$DMKey = KDF(MTCdID, MTCsID, MTCdNonce, MTCsNonce, UEID, PSKey)$$

MTCs then computes and sends to the UE:

M3 =
$MTCsID||MTCdID||MTCsNonce||UEID||[MTCsInfo]||MTCsMIC||DMKey$

MTCsInfo can be useful if the MTCs wants to select a particular KDF, or to limit the actions of a user towards the specific MTCd in terms of usage type, duration and DMKey expiration/revocation. This is left as an open topic for MTC server's policy.

No specific KDF is advocated except that general security good practices should be followed. Therefore, the MTCs should be able to select the best option for the related MTCd but the KDF recommendations in [76] are strongly suggested to be followed.
Message 4: When the UE receives **M3** from MTCs it extracts, stores and deletes DMKey from the message. By receiving DMKey, the UE has the implicit indication that the MTCd has been successfully authenticated. It then sends to the MTCd:

M4 = $MTCsID||MTCdID||MTCsNonce||UEID||[MTCsInfo]||MTCsMIC$

Upon reception, the MTCd uses these elements to verify MTCsMIC. If the verification is successful, the MTCd authenticates the MTCs and the mutual authentication process is complete. It then computes DMkey:

$$DMKey = KDF(MTCdID, MTCsID, MTCdNonce, MTCsNonce, UEID, PSKey)$$

After the protocol is executed, the MTCd also implicitly authenticates UEs as they now both share a DMKey, a shared secret key that can be used to derive further confidentiality or integrity protection keys. This solution can act as the underlying mechanism for ProSe, using the methods described in [73] to derive further keys.

6.6 Authentication Protocol Extension

In the considered scenario, another UE in proximity of a group of MTCd or a GW is now added. A user needing to connect to one or more MTCd can request the connection establishment to their ProSe links, UEs or GWs. If it doesn't have any ProSe pair, ProSe discovery request can start as defined in [31]. It is advocated that ProSe Direct Discovery and Direct Communication concepts [31] can be used between an UE and a GW for cooperation and coverage extension. After this connection is established, the UE/GW tries to access the MTCd the user requested. If it succeeds, the protocol proposed in the previous section can be executed, with one more actor, the second UE or a GW. Fig. 17 illustrates the extended version of the protocol. In addition to the messages defined and explained in section 6.3, we now add two more messages, **M2** and **M6**. They are however the same as described before, only the new UE or GW is forwarding them to the correct destination. This is a simple protocol extension proposal but yet, it can be extremely effective. Links can be created involving several UE or GW nodes, creating longer routing paths and exploring then the full potential of D2D connections, including the use of the ProSe standard to leverage the coverage of the 5G network.

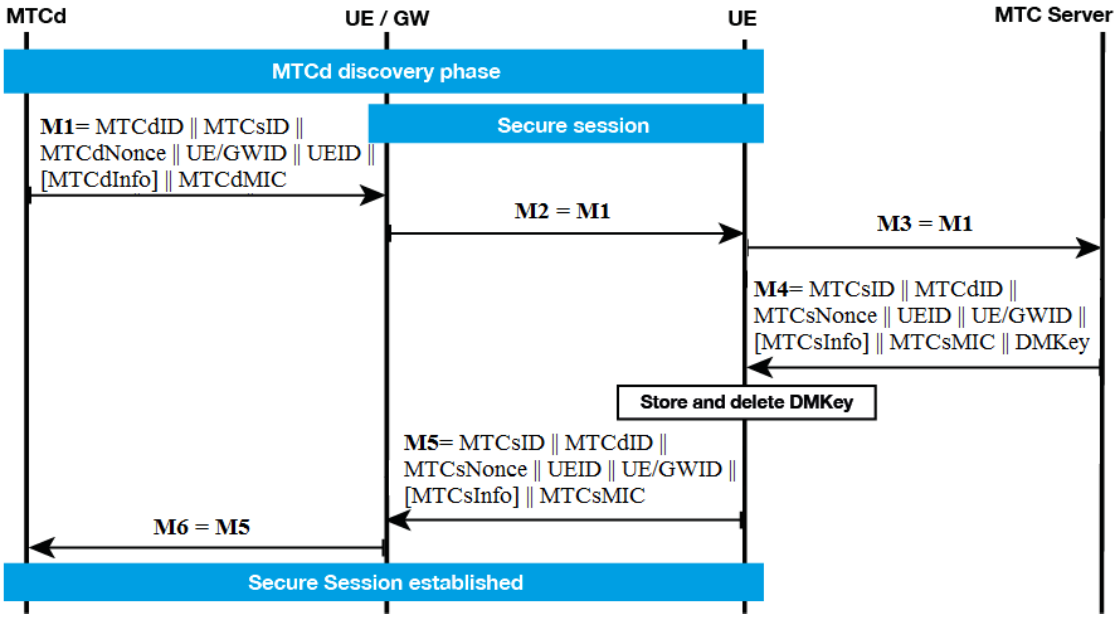


Figure 17: Extended authentication protocol

6.7 Security evaluation and analysis

In this section, the security properties of the presented solution are evaluated. This security analysis is built based on Dolev-Yao threat model. The proposed protocol is fully compliant with all the security requirements of 3GPP [73]. A trust relation is built between all elements as the protocol runs.

ProVerif [77] was used to test the authentication and secrecy properties of the protocol. After representing it with this tool, the secrecy of DMkey was queried, as well as the authentication of the MTCd by the MTCs and vice versa. Positive results for all 3 queries were obtained. The secrecy and mutual authentication properties of this solution were therefore proved. The metrics for security evaluation are symmetry of the keys, if the scheme is probabilistic or deterministic and if it is server assisted. Confidentiality and integrity of the messages, authentication, authorization, freshness of messages and resilience to attacks are also evaluated. This solution uses a symmetric key scheme to account for limitations of MTCd. It is deterministic by design and relies on the MTCs to assist the key establishment. Messages exchanged can have confidentiality and integrity by means of the shared DMKey. The authentication is mutual and explicit for the MTCd-MTCs pair by verification of the MICs. It is mutual for the pairs UE-MTCs. They trust each other as their secure channel was previously established by the 3GPP system. It is mutual and implicit for MTCd-UE pair, the moment their symmetric keys are established, because when MTCd verifies MTCsMIC, it means the MTCs trusted the UE. Authorization can be checked for all participants. MTCs is responsible for authorizing MTCd as per its own policy and to check with the Core Network if the UEs and GWs are authorized to establish the connections. MTCdNonce and MTCsNonce miti-

gate replay attacks. Finally, this solution is resilient to attacks due to the use of pairwise keys and therefore compromising one node does not compromise the whole network.

6.8 Performance

The security metrics and performance of this solution are now evaluated and they are compared with works presented in [47, 48, 78]. The works are evaluated as deterministic or probabilistic for key establishment procedures. In [47], a probabilistic scheme is proposed and probabilities for key establishment are presented. The authentication is via a coalition of UEs in [48] and in both [48] and [47] servers are not involved. In [47], the server is also not required for communication establishment. In this sense, these proposals are described in this section as providing implicit authentication. In both the presented solution and in [78], there is support for authorization and explicit mutual authentication between MTCd and MTCs. In this proposal, MTCdInfo and MTCsInfo are reserved to use to limit the access for certain data or application type, access time, or any other information that suits their needs or policies. Finally, it is shown that the Server is a necessary actor in the proposal presented both in this section and in [78] so that the MTCd are authenticated and the protocols executed. Table 7 summarizes the qualitative comparison assessment of the security features.

Table 7: Security metrics comparison

	Key establishment	Authentication	Authorisation	Server assistance
[48]	Deterministic	Coalition of devices implicitly autheticates the new member.	Not specified	Not needed
[47]	Probabilistic	Implicit authentication by having a common key	Not specified	Not needed
[78]	Deterministic	Explicit and mutual for MTCd-GW	Supported	Needed
Sec.6	Deterministic	Explicit and mutual for MTCd-MTCs Implicit and mutual for MTCd-GW	Supported	Needed

Performance is evaluated in terms of number of messages necessary for key distribution, computational effort required to run it and memory requirements. This proposal needs four messages to be executed. The cooperation scheme adds two more messages, but they are simply forwarded from one UE to another UE/GW, without extra computations. The MICs and nonces provide explicit mutual authentication and mitigate replay attacks. MTCsInfo allows MTCs to be able to choose a suitable KDF and to restrict the usage of the MTCd, as per its policy. Therefore, it is concluded that the elements in the

messages are the minimum possible to guarantee these security properties. Symmetric key cryptosystems are well suited for resource constrained MTCd. The KDF executed in the server side and the key delivered to the UE, eliminates the need for the latter to make computations. The MICs are calculated in a standard, recommended way [74, 75] and therefore, the computational cost is normal. The MTCs can choose the KDF from the recommended ones in MTCsInfo.

Table 8: Performance comparison

	CPU usage	Memory usage	Messages Tx/Rx
[48]	**High** Assymetric cryptosystem	**High** Use of certificates	Messages exchanged on demand: minimum 3
[47]	**Low** Symmetric cryptosystem (Computes key verification)	**High** use of key rings (higher the number of keys, higher the probability)	Messages sent periodically: every 10ms
[78]	**Low** Symmetric cryptosystem (Computes MICs and KDFs)	**Low**: 1 PSKey, 1 Group Key, 1 Derived Key (per pair MTCd-server)	Messages exchanged on demand: 4
Our	**Low** Symmetric cryptosystem (Computes MICs and KDFs)	**Low**: 1 PSKey, 1 Derived Key (per pair MTCd-UE)	Messages exchanged on demand: 2

This can be very useful for the MTCd as the MTCs can, for example, select a suitable KDF to generate DMKey. Therefore, it is concluded that the computational costs and computing power as minimum to guarantee robust security features.
As for memory, some bytes are needed to store keys. The nonces require some more bytes to store previously used values but in very constrained MTCd, they can be replaced by counters. The MTCs needs to maintain a database linking MTCd IDs with the PSKey, DMKey and nonces but memory shouldn't be a problem at a server level. The power consumed in communications can be reduced after the D2D connection is established. It can reduce congestion risk if the interaction is with several MTCd at the same time. The scalability may be affected for a big number of MTCd but as the protocol relies on proximity, this is not foreseen as a problem.
This proposal is compared with the works mentioned in Sec. 4. The performance is evaluated in terms of the main energy spending contributors: CPU usage, memory usage and numbers of messages Tx/Rx. The proposed protocol requires less messages exchanged, provides all the modern security features and complies with 3GPP standards [73]. To better demonstrate the benefits for MTCd in power savings, both after

the D2D connection is built and during its establishment, these metrics are regarded from the point of view of the MTCd. It is worth to mention that, to the best of my knowledge, no other proposal for direct MTCd-UE communications was found in the scientific literature. In [78], a protocol to authenticate a group of MTCd is proposed. To make a proper comparison, it is assumed their protocol is authenticating one MTCd only. Table 8 summarizes the comparison of the four proposals in terms of the mentioned performance.

6.9 Remarks on the Security Protocol

In this section, a protocol for authentication and establishment of secure sessions between UEs and MTC devices without any prior trust relation was proposed. All cryptographic systems used are lightweight to account for resource constraint devices. The solution has great potential for energy, bandwidth and latency benefits. It also introduces means for coverage extension taking advantage of the users' mobility, having ProSe as an underlying mechanism and providing therefore, an extension to existing standards.

7 Real Time Dynamic Security for ProSe in 5G

7.1 Introduction

Device-to-device (D2D) communications have been studied for some years now due to their potential on improving network functions and performance such the potential to an increased throughput, energy savings and efficiency or the potential to reduce delays. In cellular networks, 3GPP has been developing the standard that defines direct UE communications, the ProSe standard.

ProSe is expected to bring many new services, features and applications. The new applications to come can be essentially divided into 2 main categories, Public Safety & Critical Communications and Commercial Communications. Public Safety & Critical Communications have the goal of providing reliable communications in PPDR scenarios. In these, the network may not be available either for a short time due to some unexpected problem or for a longer time (maybe even permanently), in areas where seamless connectivity is difficult to provide, either in land or sea. This standard was first introduced with the objective of replacing old emergency communication systems for police, firefighters and medical personal like 3GPP's Terrestrial Trunked Radio (TETRA), a very old emergency communications system. On the other hand, commercial applications may relate to many aspects of daily life. Proximity social networking, interaction with smart cities or vehicle-to-everything would all fit the commercial use cases.

One of the objectives of operating without network supervision (or outside coverage) was achieving seamless communications in PPDR scenarios while under supervision, the main goal was first to offload communications away from macro BS, releasing resources for other users and saving energy at the BS level. The two factors were the starting point for ProSe's development but at the present day, researchers and industry see many other advantages of using it, mainly due to the envisioned possible applications. Therefore, several application areas for D2D are under study. They include vehicular networks, autonomous machines, the IoT, Public Safety communications and the proposed direct MTC-UE communications.

There are different possible security operating modes for UEs in ProSe [31] depending on its network coverage status, its role as a Public Safety UE (PSUE) or interaction type. UEs can interact in direct One-to-One (1:1) or One-to-Many (1:M) communications both types have different security mechanisms and rules during the establishment, active phase and termination of a connection. The rules and mechanisms include requirements in terms of cryptographic objectives, signaling procedures, key management, cryptosystems and cryptographic primitives. This results in a complex combination of possibilities for UEs in terms of security contexts. ProSe does not deal with by not specifying how to implement security policies. A very modest attempt is done with the inclusion of a key ID in the header of each Packet Data Convergence Protocol (PDCP) packet sent so that from it, the receiving UE can locally identify a root and derive further cryptographic keys. Although this is enough for cryptographic algorithms to work, it does not provide any information to UEs on security contexts and policies, which raises several problems, such as:

- ProSe UEs don't have a set of security rules (a context or a policy) designed and administered by the Core Network (CN), making ProSe networks generally less secure. UEs without policies can promote changes in security settings, leaving connections vulnerable to malicious devices that can try to enforce lower security levels or even no security (downgrading attacks);
- UEs that unexpectedly loose network coverage are automatically unable of starting new ProSe communications
- Keys for group communications are provided with expiry timers. Not provisioning PSUEs or UEs may lead to expiry of all provisioned keys in extreme PPDR scenarios or long missions, possibly disabling UEs from communicating, stopping them from delivering emergency support as well as from providing coverage, routing or backhauling services;
- The security context of a connection cannot be changed unless heavy signaling messages are exchanged (1:1 case) or root keys expire (1:M case) [79]. As a consequence, connections either require extra signaling or they always need to maintain the same security;

Therefore the need for a solid security bootstrapping mechanism id identified where the mechanism can solve these problems and provide security policy information to the UEs and PSUEs, anticipating any new connection establishment, temporary or longer loss of coverage like PPDR scenarios.

There is also the need to allow for changes in the security settings of a connection, but the required signaling between UEs or from UEs to the CN increases energy consumption and can create bottlenecks. Changing a security context make ProSe networks and connections less rigid, more intelligent and allow for security to adapt dynamically to different conditions. There are different reasons that call for the introduction of dynamic changes in security of a connection, as opposed to an established, fixed context. Namely:

- UEs that use energy harvesting hardware may have a surplus of energy due to having a full battery at the same time energy can be harvested. More security services may be executed not to waste the surplus energy while increasing network security.
- An Intrusion Detection Systems (IDS) monitors the network and can send alerts so that UEs increase security measures in their connections. On the contrary, if trust levels are higher, the IDS may reduce the applied measures;
- There are different types of data transmitted in a network. For e.g., some data can be more sensitive due to containing users' personal information or requiring 100% accuracy on the data transmitted. To protect it, different security mechanisms are needed;
- Executing less security mechanisms might be an effective way to implement an energy saving strategy;

Therefore in this section, the presented problems are addressed taking into account the ProSe standard and the benefits related to dynamic security. Specifically, the presented contributions are 1) the specification of security levels based on cryptographic services, 2) mapping of those levels based on the different operation modes and packet types in ProSe, 3) the introduction of a security context change method with minimal cost and overhead and 4) the design of a list of security parameters and their provisioning to UEs, including its inclusion in Multimedia Internet Keying (MIKEY) protocol. To the best of my knowledge, this is one of the first approaches available in the literature for dynamic security and the first one in ProSe.

7.2 ProSe

In this section an overview over the ProSe standard is presented. At its core, ProSe defines the rules and procedures for the D2D communications between UEs with or without network supervision.

7.3 Scenarios

Fig. 18 depicts different communication possibilities. UEs can use ProSe in coverage of a BS as marked with labels 1 and 2, or out of coverage, signaled with labels 3 and 4. In 5, a UE provides relay service extending coverage to remote UEs. This mode of operation is called UE-to-Network relay. D2D links can be exclusive between 2 UEs or 1:1 type (labels 1, 4 and 5), or between a group of UEs or 1:M type, where packets are sent in multicast (labels 2 and 3). Outside coverage, only PSUEs are able use ProSe, subject to CN authorization. In there, the ProSe Function (PF) is responsible the manager of all the security related information. The element responsible for the key management is the ProSe Key Management Function (PKMF).

7.4 Communication phases

The interface between UEs is called PC5. Before any communication in PC5, UEs must be aware of their neighbor nodes. This is accomplished by Discovery messages of model A or model B. In the first case, UEs announce their presence in a "I am here" model. In the latter, requests are sent in "who is there?" and/or "are you there?" model [31]. The latter, where a specific user is targeted to be found is an example of privacy leakage risk. Target UE's information like its ID, regular location, services used and other private information can be extracted from Discovery messages, imposing the need to apply different security measures on different messages. With the discovery process complete and after security is established, UEs communicate directly. All messages that serve call management purposes are Control Plane (CP) messages. Authentication procedures or link control messages are CP examples. The actual data that users exchange is User Plane (UP) data. The security requirements for these three main groups are quite different. They differ on message type, network status of a UE (in or out of coverage and relay) and on communication type (1:1 or 1:M). For example, UP data can be encrypted

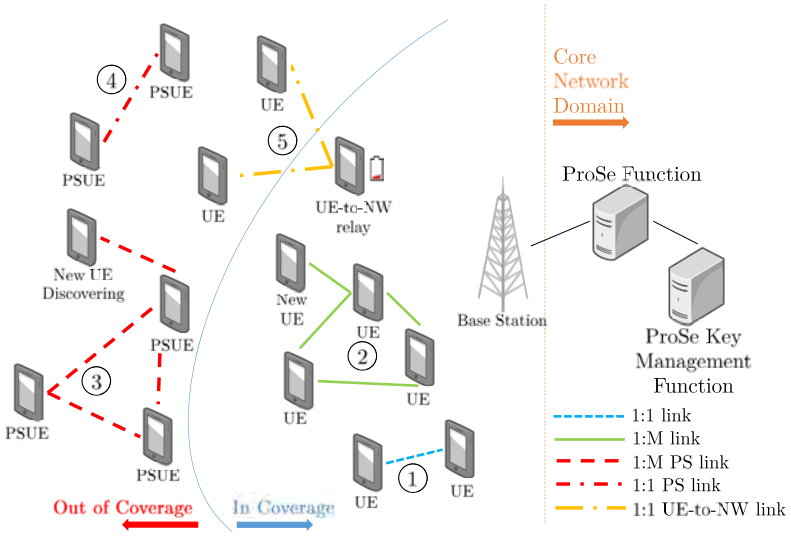


Figure 18: Overview of ProSe

or not but it shall not be integrity protected. In all cases, the security services associated with a message are configurable options. However, due to the key management policy in ProSe, the configuration is rigid and changing security context is in some cases impossible or requires extra signaling. This is detailed in the next subsection.

7.5 Key management

A security context is the set of security services for a ProSe 1:1, for a group or for UE-to-NW relay links. In all cases, the security context is linked to a root key from which further layers of keys may be derived for cryptographic purposes. Upon authorization request to the CN to use ProSe and consequent approval (Discovery, 1:1, 1:M and UE-to-NW modes), all root keys are transported to the UEs except in the 1:1 case, where UEs reach key agreement using one of several asymmetric methods.

Discovery messages are protected using directly the keys that were provisioned. If the PKMF does not deliver a key, e.g. encryption key, it means that specific service is not meant to be used. In case there is a need to change security context for any of the reasons mentioned in section 7.1, extra signaling is needed so that the UE has an encryption key delivered as shown previously in Sec. 2.

In 1:M, root keys with different expiry timers are transported via MIKEY protocol to UEs and only the one with the earliest expiry time is valid. From the root key, further layers of keys are derived. In this case, as the UE always uses the key that expires first before changing, in case there is a need to change security context like mentioned in section 7.1 it will not be possible before the associated earliest timer expires.

In 1:1, a root key is agreed after an asymmetric direct authentication procedure. From the root, one key is derived in the next layer. The ID of this key is sent in the PDCP header of all packets sent and is used to identify a security context on the receiver side. As there can only be one key in the second layer of keys, and in case there is a need to change security context, a rekeying process is necessary along with renegotiation of capabilities and possibly a new, energy expensive mutual authentication procedure.

In UE-to-NW, keys are also transported to both relays and remote UEs via a ProSe protocol instead of mutual authentication and key agreement, while both are still in coverage. But although the root key establishment procedure is different, the remaining of the security is the same as in the 1:1 case and therefore, so are the problems.

Excluding the 1:M case, where the security context may change via timer expiry, rekeying is mandatory to make changes in a set of security rules. The result is generally increased signaling and security overhead or the penalty of having a static context (not being able to efficiently use energy from harvesters, to respond to IDS systems and to protect certain types of data). It results as well in an inability from the CN to impose security policies on UEs. Not having means to implement and enforce a security policy on ProSe UEs creates a general exposure to downgrading attacks because UEs are basically allowed to negotiate security capabilities at will. This vulnerability can affect both 1:1 and group communications and therefore, it can compromise a big number of devices in the network at the same time. To address these problems, in the next sections, a structured organization for security levels with different security services, a method to change the security of a connection reducing general signaling and another to provision and enforce a security policy on UEs are all presented.

7.6 Security levels

Four security levels are defined based on cryptographic services. Table 12 depicts the services used at each level.

Table 9: Allowed security levels per message type

(a) Security services

Security level	Data Authentication	Integrity	Confidentiality
Level 4	✓	✓	✓
Level 3	✓	✓	
Level 2			✓
Level 1			

(b) Allowed security levels

Network status	Packet type	Allowed Levels
Out of coverage/ UE-to-Network/ In Coverage	Discovery	Lv1 - Lv4
	CP	Lv1 - Lv4
	UP	Lv1 - Lv2

Hence in level 1, no security service is provided. In level 2, only confidentiality is used. In levels 3 and 4, both data authentication and integrity protection are used. The property results from the allowed integrity protection algorithms in 3GPP systems that tare all keyed. Level 4 provides confidentiality as well.

After in depth inspection of the security requirements in [73], different policies can be inferred for each network status and packet type. They are valid for both 1:1 and 1:M communications. Table 9b summarizes the allowed levels for each combination [73].

7.7 Security context change

In order to change a security context, UEs do not need to start a rekeying process. Instead, a UE initiating the context change process selects the NW status that wishes to change. It can be the current status only or all the three options (because of e.g. loss of coverage predicted). For each NW status that will be updated, Discovery, CP and UP security levels and algorithms used in each security service need to be updated. As there are 3 NW status and 4 levels, 2 bits are enough to represent each. One extra byte is included to inform the algorithms used. Figure 19 depicts the information element (IE) needed to change security context and its size.

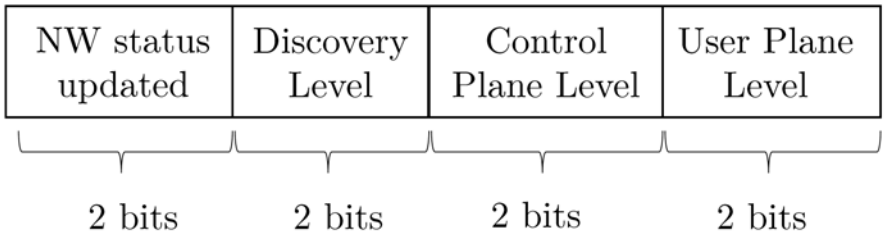


Figure 19: Security context change Information Element

As a result, the security overhead for this operation is between 2 and 4 bytes, varying on how many contexts the UE is trying to change. This IE can be sent over a pair *Security Mode Command* and *Security Mode Complete*, not deviating from ProSe standard and previous cellular networks standards, where this pair is also used for security purposes. The first message informs about the context update while the second serves as confirmation or rejection of that context. Rejection may come, e.g. due to non compliance with PF policy. This is discussed in the next section.

Compared with ProSe's root key rekeying process [73], the security overhead and the transmissions needed are substantially reduced for the 1:1 and UE-to-NW relay cases. In 1:1, the rekeying involves four steps. A *Direct Rekey Request* message followed by an entity authentication procedure with subsequent key agreement and a pair *Security Mode Command* and *Security Mode Complete*. In these messages, several parameters related to the rekeying and sent by UEs. In table 10, a comparison is made between

our proposal and a rekeying process for a security context change in terms of number of messages and IE data size.

Table 10: Messages and IE size comparison

	1:1		1:M		UE-to-NW	
	Msgs	IE	Msgs	IE	Msgs	IE
ProSe Rekeying	3+Auth	11+Auth (bytes)	N/A	NA	5/6	15 (bytes)
Context change	2	2-4 (bytes)	2	2-4 (bytes)	2	2-4 (bytes)

In the UE-to-NW relay case, if the remote UE triggers the rekey of the root key, five messages are needed. If the relay triggers it, six messages are needed. These include messages to the CN. The IE data size is also shown in table 10. For 1:M, there is no rekeying process defined and therefore, it is not possible to make a direct comparison. However and for this case, this proposal offers a solution for the rekeying problem.

In the table, neither IEs exchanged with the CN nor authentication data size are accounted for in order to compare only the direct impact on the security overhead on the UEs and because there are different authentication methods that can be used. As shown, this proposal introduces significant benefits by reducing communications, security overhead while allowing a faster change in security context as well as the reasons mentioned in section 7.1. However, just like in the rekeying process, signaling to the CN is needed so that PF policies can be enforced and UEs operate in accordance to them. Therefore, to complete the solution, a method for bootstrapping security parameters is detailed in the next section.

7.8 Bootstraping ProSe

In order to provide means to the PF to have security policies enforced on UEs, information related to it needs to be provided to UEs. A method to bootstrap it is proposed before any kind of communication takes place. The purpose is to provide information to UEs so that security contexts can change according to the proposal from the previous section, with reduced signaling and security overhead and always under the policies defined by the PF. To that purpose, a list of parameters to bootstrap in UEs is presented and the bootstrapping timing for 1:1 and UE-to-NW cases is detailed. In the case of 1:M, a bootstrapping method for MIKEY is also detailed.

7.9 Parameters

The list was designed to address ProSe's requirements, restrictions and modes of operation described in [73] and is presented in table 11.

First parameter signals authorization to work as a Public Safety UE (PSUE). This functionality needs to be signaled to UEs and may be extremely important enabling non PSUEs e.g., in PPDR emergency scenarios, which cannot be achieved by any hardcoded

Table 11: Parameters bootstrapped in UEs

Type	Parameter
0	Public Safety UE authorization
1	UE-to-NW relay authorization
2	ProSe Out of Coverage authorization
3	UEs authorized in Restricted Discovery
4	Min. Sec. Level override flag for ProSe
5	Allowed Sec. Levels for Disc
6	Allowed Sec. Levels for CP
7	Allowed Sec. Levels for UP
8	Allowed Algs. for Disc. messages
9	Allowed Algs. for CP messages
10	Allowed Algs. for UP messages

method. Parameter 1 authorizes a UE to act as a relay extending coverage to remote UEs. The value used provides information to the UE if it can be a remote UE only, a relay UE only or both. Parameter 2 authorizes (or not) a UE to use ProSe when out of coverage. As the UE is being provisioned for ProSe, it is already authorized to directly connect while in coverage. In the special case of Restricted Discovery, not all UEs are allowed to be discovered by a discoverer UE. A list of UE IDS or other elements that can identify discovered UEs is therefore transmitted via Parameter 3. Parameter 4 is a flag that indicates whether the security levels being used can be overridden. This may be a fundamental functionality for PSUEs as a communications enabler in PPDR scenarios, where incompatible security capabilities that block communications can exceptionally be surpassed. It can also be used to bypass security as an extreme energy saving strategy. Parameters 5 to 7 define the security levels the PF allows for each message type (i.e. its security policy for security services). This helps preventing against downgrading attacks and keeps some freedom of choice to UEs to manage the security of their connections dynamically but always inside the imposed limits by the PF. Finally, Parameters 8 to 10 define a list of allowed algorithms to be used for entity and data source authentication, confidentiality and integrity protection. In the list, that should be kept open to further additions, should figure at least 3GPP standard algorithms like the EIA and EEA families.

7.10 Bootstrapping phase

Whether it is 1:1, 1:M or UE-to-NW type communications a UE will use, authorization is the first and fundamental part of the process. In 1:1 and UE-to-NW types, immediately after authorization is granted, parameters are provisioned from the PF to the UE. In these cases, the bootstrapping of the proposed parameters happens at this moment.

For the 1:M case, if a UE is authorized to use ProSe, MIKEY protocol is executed between the UE and the PF so that root keys and their expiry timers are delivered to

UEs. Due to the use of MIKEY for group communications, it is proposed to include the bootstrapping parameters in the MIKEY messages. In MIKEY specifications, several payloads are defined so that MIKEY itself can be integrated with other protocols. In the case of security, a specific payload is defined in defined in [6] and it is depicted in Fig. 20.

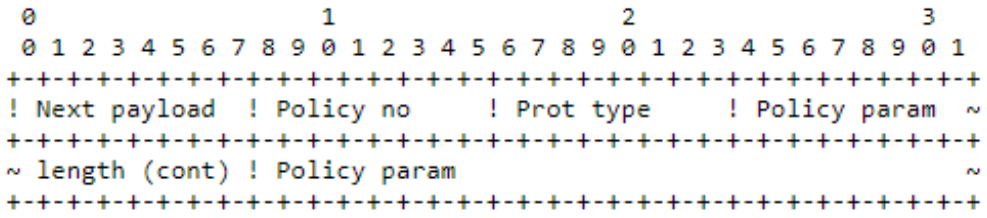


Figure 20: Security Policy Payload [6]

From it, the relevant parameters to be detailed are *Prot type* and *Policy param*. The first defines the target security protocol which will use the parameters transported by MIKEY. At the time of writing this work, only two protocols are defined in the standard using values 0 and 1. ProSe would use value 2. The *Policy param* is further built up by a set of Type/Length/Value (TLV) payloads. The parameters provided in table 11 are included here. It is important to note that these parameters are of variable size, depending on each protocol's needs. Therefore, the parameter definition and the presented solution is fully compliant with MIKEY.

With the presented bootstrapping mechanism, UEs can be provisioned with a security policy defined and maintained at CN level. From the three presented processes, the one that occurs first can be used to provision all information, further reducing the information amount sent during provisioning UEs when compared to sending information in three different occasions, provisioning for 1:1, 1:M and UE-to-NW. It also solves the problem of reaching UEs that can be suddenly out of coverage for a short period or in PPDR situation, that might take much longer.

7.11 Benefits of the solution

The benefits of the proposal in this work are now discussed. Starting with the introduction of different security levels has several advantages. Common security levels are standardized for all connections. This facilitates the introduction and the tasks of IDSs by giving it a tool to change security settings in real time for specific communications, i.e. specific UEs, groups, applications, domains or geographical areas, e.g. under specific BSs. It allows for flexible security mechanisms where specific cryptographic primitives can be applied to certain message types only. It allows for the adoption of energy saving strategies on the UE side and in case of energy surplus due to the presence of energy harvesting hardware, it provides means to increase network security.

The use of the pair *Security Mode Command* and *Security Mode Complete*, which comes from previous 3GPP releases and is also used in ProSe, proves to be enough to change the security context with minimal security overhead, minimal number of messages and requiring no signaling to the CN and thus, using no resources beyond the BS. Our solution carries a maximum of 4 bytes and requires only an acknowledge as a response. The numerical details are in table 10. In the case of 1:M, it provides a solution for the context change problem.

Bootstrapping the parameters in UEs has also several advantages. It provisions the UEs with a set of levels of security, allowed tasks and algorithms for different call procedures defined in [31]. Hence, connections change their security levels only according to the PF policy which mitigates the risk of downgrading attacks. It allows network operators means to set security policies to all connections represented in Fig. 18. For all the communication types, UEs participating in calls are informed about the security policies. Consequently, deviations from allowed security behavior can be more easily observed and reported, improving the system's monitoring and intrusion detection abilities, making it more secure in a general way. It can be useful as well for solving incompatibility issues. If a UE is not able to execute a more recent version of e.g. an encryption algorithm, will still be able to participate in a group call by negotiating inside the list of the allowed ones, with no need for further requests to the CN. This is especially important in the out of coverage cases. Finally, UEs in PPDR scenarios need to have security information bootstrapped while in coverage under the penalty that they won't be able to use ProSe, or cannot offer relaying and backhauling services.

7.12 Remarks on Real Time Dynamic Security

In this section the concept of dynamic security levels based on security services for ProSe communications was introduced, an inexpensive method for security context change in real time and a proposal for bootstrapping security parameters in UEs, totally compliant with ProSe standard and MIKEY protocol. The concept creates a common structure for all UEs and it allows for efficient energy saving and security increasing strategies. The offres a solution for the problem of out of coverage UEs enabling ProSe even in long time PPDR scenarios.

8 Optimal Security Context Selection in IoT Radio Links

8.1 Introduction

One of the most highly underestimated sources of energy consumption in wireless networks is cyber security. There is no shortage of works in the literature studying the consumption of security primitives and protocols in particular devices, quantifying this energy consumption [7, 8, 23–28, 80, 81]. Several works that are referred to in the next section are also found studying collaborative approaches or mechanism improving strategies for security that aim at reducing energy consumption or computational complexity. Although some of these approaches have proven to reduce energy consumption, their contribution to reduce it is limited to the extent that they aim at still maintaining the security functionalities.

One of the most challenging problems to address in 5G networks is indeed energy related. The predicted exponential growth of the number of communicating devices and data [71] is predicted to increase energy consumption to unprecedented values [82]. In contrast with this demand, the hardware trend is to reduce battery capacity and introduce EH hardware, further limiting batteries' role as energy storage units or even replacing them completely, if energetically viable [83] [39] [84]. Therefore, increasing energy efficiency and ensuring network survival becomes a major challenge, particularly in the case of the IoT, where the number of resource constrained devices connected to 5G networks is also expected to grow dramatically [71].

The very definition of network lifetime in sensor networks kept mutating over the years. Work presented in [85] surveys eleven different definitions for network lifetime. Different metrics are used to build all eleven definitions but it is clear that they all relate to the energy depletion of one, several or all the nodes in the network. These works also show that network lifetime starts to decrease more rapidly after the energetic death of one node [61], which underlines the importance of addressing the energetic survival of single nodes.

Enabling security features induces energy consumption mainly due to security overhead data sent in the air interface and processing tasks. Works in the literature profiling energy consumption of wireless devices with constrained resources show that the energy consumption of the communications module in a device can be as high as 14 times that of the processing tasks [46], indicating the transmission module as the main source of energy consumption, and with a big gap in relation to other device components such as sensing and processing blocks.

In standards that run on top of IEEE 802.15.4 based radios like e.g. Zigbee, security features can be provided at the application, network and data link layers. [86, 87] The use of security features in each of these layers creates security overhead that is appended to payloads as they descend towards the physical layer, where the energy consumption is measured. Considering merely two of the most widely used security features, confidentiality and data authentication, the security overhead at the physical layer can be up to 63 Bytes in a frame of 127 Bytes size, the maximum size defined in IEEE 802.15.4 standard. This represents a tremendous energy burden on communicating devices.

On the other hand, IoT networks today rely on security features to try to guarantee normal operation under the threat of possible malicious actions making them fundamental features for normal operation, if malicious actions are ongoing which only happens on very small time windows and specifically targeted networks. One can argue therefore that with no such threat present, the energy consumed with cyber security contributes immensely to degrade the energy efficiency of a network and after quantifying it like was just done, it is concluded that constitutes a very underestimated consumption source.

The reasons found that motivate exploring the trade-off between security and energy are summarized below, considering data authentication and confidentiality as security features:

- recent developments in IDS and Intrusion Prevention Systems (IPS) in particular, provide good means of malicious action detection and deflection (e.g. with honeypot technique), creating an additional protection layer that can assure with a high level of confidence whether a cyber threat is active or not; [88, 89]

- in applications where data reliability is the main concern, packets sent without security features are more important than discarded packets due to insufficient energy;

- some of the data circulating in a network holds little to no value of being accessed. A simple example is temperature readings from sensor networks installed in street light bulbs, reducing the importance of providing confidentiality;

- in very difficult to access or very isolated physical perimeters, the importance of data authentication is reduced due to diminished radio range to perform, e.g., spoofing attempts;

- securing all layers in a protocol stack provides maximum protection but trading security for energy in one or two layers does still provide data protection;

- nodes routing traffic that decrease security usage make part of an E2E link more fragile, but only in a segment of that link.

In this section, the aim is to address this very challenging trade-off security/energy after recognizing security features as for being both important for normal network operation and protect against malicious actions and for being a critical source of energy consumption. The predictions for ICT energy consumption worldwide [82] call for efficient energy reduction mechanisms. The problem is addressed by applying machine learning techniques to learn optimal strategies that target at maximizing both aspects of the tradeoff. To the best of my knowledge, this work is the first attempt to discard security measurements as a mechanism to ensure device and network survival and the first work implementing energy aware security features in D2D communications.

8.2 System model

A MTC device is considered to be communicating with another network element where the communication channel used is discretised into time slots of equal duration. At each time slot n, a packet arrives at the Data Link Layer from upper layers and needs to be transmitted by the MTC device to their intended receiver. The transmission of each packet has an associated energy consumption. The energy required for the transmission is taken from a battery installed on the device. The device is equipped with an energy harvesting hardware that collects energy from the environment, converts it into electrical energy and stores that energy in the battery. The packets are then assigned with security features to protect them and transmitted.

8.3 Battery model

The MTC device has a battery of size $b_{\max}$ and is divided in equal size parts where each of those parts is defined as an energy quanta. At any time slot, the battery level is $b_n \in \{0, .., b_{\max}\}$, where b_n represents the amount of energy available in the battery at the beginning of time slot n, measured in energy quanta units.

8.4 Energy harvester model

The energy harvester supplies a value of h_n energy units to the battery during time slot n, each of those units corresponding to an energy quanta. At the end of time slot n, if the energy harvested plus the energy available in the battery is greater than what the battery can store, the extra energy is lost, i.e., if $b_n + h_n \geq b_{\max}$ then $b_n + h_n = b_{\max}$ or $b_{n+1} = b_{\max}$.

The energy harvested in time slot n is governed by a random variable (R.V.) H_n, where h_n indicates the actual amount of energy that is harvested in time slot n. The harvested energy is assumed to be i.i.d. across time slots and to be governed by the probability mass function (PMF) $p_H(h_n) = \text{Prob}[H_n = h_n]$ with $h_n \in \{0, 1, \ldots, h_{\max}\}$.

8.5 Security features

At the beginning of each time slot n, a security context is chosen and applied to the active communication session and used during that slot. It is assumed that the communicating devices are already authenticated and share cryptographic keys that allow to communicate over a secure channel. In this work, a security context refers to a combination of security features or services that will be used to protect the data transmitted by the device. The considered security features are confidentiality, integrity protection and message authentication. Most of the latest, recommended and widely used algorithms that provide integrity protection are keyed [90], i.e., a pre-agreed or pre-distributed cryptographic key is used as input for the algorithm and therefore, when integrity protection of a packet is provided, message authentication is also provided. This results in four considered security contexts and a security level is attributed to each one. Table 12

Table 12: Security context per level

Security level	Data Authentication	Integrity Protection	Confidentiality
Lv4	✓	✓	✓
Lv3	✓	✓	
Lv2			✓
Lv1			

summarizes the security features considered for each context/level as well as the security overhead they require, discussed below in sub-section 8.7.

The considered set of security contexts is sufficient to describe and cover all possibilities in terms of the two considered security features in this work, that are used after security establishment takes place between two communicating devices.

The set of contexts presented is compliant with the one defined in IEEE 802.15.4 standard. There, the considered security features are the same as considered in this work, only that the standard allows for the use of different cryptographic key sizes, forming a set of eight security levels instead. For compatibility with the standard reasons, it is considered that the mentioned features to be provided at the MAC Layer.

8.6 Packet arrivals and transmissions

The packets arriving at the MAC layer are encapsulated and become the MAC payload. They are considered to have variable size and denote that as d_n Bytes. A MAC header of variable size and a MAC footer will be appended to the payload that will then be transferred to the physical layer before the actual transmission.

8.7 Energy consumption

One security context will always be in use during each time slot and will influence the energy consumed for the transmission of each packet. Packet size in this work refers to the payload size that arrive at the MAC layer and is denoted by d_n. The security level applied to the packet dictates the energy consumption of the transmission of that packet due to the extra Bytes related to security features that are appended to this payload, increasing its size. These extra Bytes constitute the transmitted security overhead, denoted by ζ.

The security context/level adopted in time slot n is denoted as c_n and taken from a fixed set, i.e., $c_n \in \{1, 2, 3, 4\}$, where the value of c_n corresponds to the security level with the same number. Fig. 21 illustrates the security overhead associated with each security level. It is considered in this work that packets are never fragmented and therefore, for that to be true in $IEEE$ 802.15.4 based radios, the maximum packet size of the payload is $d_{n_{max}} = 80$ Bytes [87] and $d_n \in \{0, 1, ..., 80\}$ Bytes.

There are several possibilities for the overhead size ζ that derive from several possibilities for security feature algorithm and cryptographic key size choice. The worst

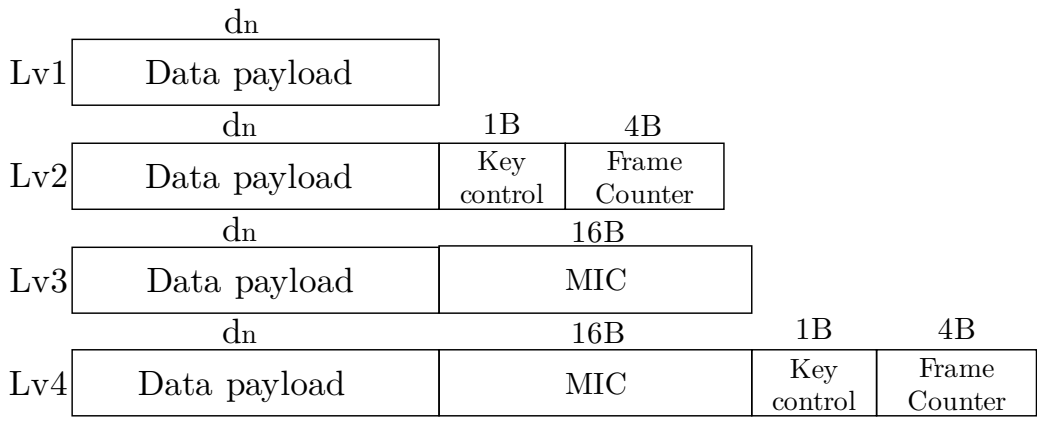


Figure 21: Security overhead for each security level

case scenario in terms of energy consumption is considered. While in Level 1 there is no security overhead associated, encrypting a packet results in a fixed appended quantity that is denoted by ζ_{conf} for frame counter and key control and that according to the standard is $\zeta_{conf} = 5Bytes$ [87]. Using data authentication adds an overhead denoted by ζ_{auth} that could be variable and relates to the MIC size used. The possible values are $\zeta_{auth} \in \{4; 8; 16\}$ Bytes [87], although it is not recommended by the National Institute of Standards and Technology (NIST) to use a MIC size inferior to 8 Bytes [91]. The total security overhead is then given by the relation $\zeta = \zeta_{auth} + \zeta_{conf}$, illustrated in Fig. 21 and the values considered in this work appended in Tab. 12.

The packet size d_n is governed by a further R.V. D_n (also i.i.d. across time slots), with pmf $p_D(d_n) = \text{Prob}[D_n = d_n]$, where $d_n = 0, 1, \ldots, d_{\max}$.

The energy consumption due to the choice of a particular security level c_n within time slot n on a packet of size d_n is thus obtained as

$$e'(c_n, d_n) = \lceil \frac{(d_n + \zeta)}{88Bytes} * 100 \rceil; \tag{7}$$

where the quantity 88 Bytes is the result of adding a MAC footer and header to the MAC payload, resulting in the maximum size allowed for these three elements together to be passed to the physical layer without causing packet fragmentation. The quantity $e'(c_n, d_n)$ is the energy consumption associated with transmitting a packet of size d_n using security level c_n. Note that $e\prime(c_n, d_n)$ is also expressed in terms of energy quanta and takes values in $e\prime(c_n, d_n) \in \{0, 1, \ldots, e\prime_{\max}\}$, with $e\prime_{\max} \leq b_{\max}$.

Considering the harvested energy and the energy consumed for packet transmission, battery evolution from time slot n to slot $n + 1$ is governed by:

$$b_{n+1} = \max\{0, \min\{b_n + h_{n+1} - e(c_n, d_n), b_{\max}\}\}. \tag{8}$$

which means that when level c_n is used in time slot n, there is an energy expenditure computed as $e\prime(c_n, d_n)$, basically decreasing the battery level in an amount equal to

$e\prime(c_n, d_n)$ and increasing it due to an harvested energy amount h_n. The system state at time n is $s_n = (b_n, c_n)$.

8.8 Problem formulation

There are various objectives in this work. If the MTC device has a low battery level, either due to the harvester not being able to collect enough energy from the environment or due to high energy consumption, battery durability should be prioritized to ensure the device's energetic survival. On the other hand whenever available, the harvested energy shall be used to increase as much as possible the security features applied to the communications. In this context, as much as possible means that whenever there is enough energy to secure packets, the device shall maximise security. The assumption that there is always enough energy is not valid for resource constrained devices in the IoT and as shown in the previous section, security features represent a significant burden in terms of energy consumption. It is therefore a very big challenge to try to ensure security features are applied to communications when there is not enough energy to do so. The optimisation of the energy-security tradeoff is one of the main goals of this work.

It is assumed that the energy harvester is chosen to suit the device's energy needs and will therefore provide enough energy most of the time. If the energy arrival is not enough, the presented system has the objective to reduce energy spending to prolong the battery lifetime while keeping security features active. In more extreme cases, the harvested energy may not be enough to suffice the consumption needs, not even if no security is applied. In this case it is considered that packets are discarded and the consumed energy in that particular time slot is zero to ensure energy survival.

Another important aspects that needs to be considered is the fact that data authentication and encryption serve two distinct purposes and cannot be compared directly. Therefore, the tradeoff between these features also needs to be considered, in parallel with the security and energy balance.

8.9 Markov Decision Process

MDP in an infinite horizon context are an adequate tool to model this communication scenario. Infinite horizon MDPs are fully defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ which correspond to the set of states $\mathcal{S}$, set of actions $\mathcal{A}$, state transition probabilities $\mathcal{P}$, reward function $\mathcal{R}$ and discount factor γ, respectively. These elements are defined in the following subsections.

8.9.1 State space

The state space $\mathcal{S}$ is the set of $s_n = (b_n, c_n)$ where $b_n \in \{0, 1, \ldots, b_{\max}\}$ is the number of energy units in the battery at the beginning of slot n, and c_n is the *current* security context, i.e., the one utilized for the whole time slot n. The maximum number of elements in the state space is therefore $|c_n| \times |B + 1|$.

8.9.2 Action Space

The action space is the set of all possible security level transitions a_n that includes decreasing three, two or one level, keeping the same level and increasing one, two or three levels, denoted respectively as $a_n \in \mathcal{A} = \{-3, -2, -1, 0, 1, 2, 3\}$.

8.9.3 State transitions

A state transition probability is defined as $p(s'|s, a) = p(s_{n+1} = s'|s_n = s, a_n)$, which corresponds to the probability of arriving at state $s' = (b', c')$ knowing that the current state is $s = (b, c)$ and action a_n is performed in slot n. Given a_n, the security context in slot n is $c_{n+1} = c_n + a_n$, where the action set at time slot n, $\mathcal{A}_n$, becomes bounded so that c_n is a valid security level. The battery evolution depends on the security level chosen in slot n (i.e., the action a_n), the packet size d_n, the battery level at the beginning of slot n (b_n), and the harvested energy in slot $n + 1$, h_{n+1}. Defining $\delta = b_{n+1} - b_n$, for any $\delta \in \{-b_{\max}, -b_{\max} + 1, \ldots, 0, 1, \ldots, b_{\max}\}$, it holds that:

$$\begin{aligned} &\text{Prob}[b_{n+1} - b_n = \delta | a_n] = \text{Prob}[H_n - e(c_n, D_n) = \delta] \\ &= \sum_{d_n} p_D(d_n) \sum_{h_n} p_H(h_n) 1\{h_n = e(c_n, d_n) + \delta\}, \end{aligned} \tag{9}$$

where $1\{x\}$ is the indicator function, being one if x holds true and zero otherwise. The transition probability $p(s'|s, a) = p(s_{n+1} = s'|s_n = s, a_n)$ is readily computed using (9), by plugging the right values of b_{n+1}, b_n and c_n.

8.9.4 Reward function

As previously described, both tradeoffs authentication/confidentiality and energy/security need to be considered. For the first challenge, a weight parameter $\alpha \in [0, 1]$ is defined. It works as a preference parameter for integrity protection and data authentication, as opposed to encryption. The weight is mapped to the security levels that provide these features and contributes to shaping the system's reward for its security performance at time slot n, that is computed according to the security level in that time slot, following:

$$f(c_n, \alpha) = \begin{cases} 1 & c_n = 4 \\ \alpha & c_n = 3 \\ 1 - \alpha & c_n = 2 \\ 0 & c_n = 1 \end{cases} \tag{10}$$

Function $f(c_n, \alpha)$ is therefore defined in $[0, 1]$ and encodes the security performance that is perceived by a designer, depending on their own preferences.

To address the energy/security tradeoff, another weight parameter $\beta \in [0, 1]$ is defined that works as a preference parameter this time for battery durability and it is

considered together with the energy available in the battery in time slot $n+1$. A function $F(c_n, b_n)$, the reward function is then defined as follows:

$$F(c_n, b_n) = (1-\beta) f(c_n, \alpha) + \beta \left(\frac{b_{n+1}}{b_{\max}} \right), \tag{11}$$

where β balances the importance of security services (first term) and battery durability (second term).

The tuning of parameters α and β is by itself a big challenge. Collaborative traffic routes, packet types, applications are just example factors that can influence the importance of one security feature over the other. On the other hand, harvester type and environmental conditions, network topology and data reliability are examples of factors that shape the importance of battery durability. It is out of the scope of this work to dwell into these factors and they are considered designer's preferences.

8.10 Markov Decision Process Model

The presented system model can then be formulated in a Markov Decision Process where the objective is to solve the Bellman optimality equation, defined as:

$$\begin{aligned} J_*(s) &= \max_{a_n \in \mathcal{A}_n} \left\{ J' \right\}, \\ J' &= \sum_{s_{n+1}} p(s_{n+1}|s_n, a_n)[F(c_n, b_n) + \gamma J(s_{n+1})] \end{aligned} \tag{12}$$

where $q_*(s,a)$ is the optimal state-action pair, $\mathcal{A}_n$ only contains values for each state that result in a valid c_n. The goal is to train an agent to learn an optimal policy, i.e., one that chooses the best action for each of the MDP states and solves Eq. 12. In case the chosen action results in $e(c_n, d_n) < b_n$, the packet is transmitted. If $e(c_n, d_n) \geq b_n$, then a reward of zero is given to the agent and no transmission occurs. As a rule, transmitting a packet is only possible when $e(1, d_n) \geq b_n$.

For any given fixed values of α and β, the Bellman's optimality equation can be solved by any dynamic programming method. The presented problem and system model have been solved using Value Iteration, i.e., an optimal policy π_* has been attained for the best possible choice of security level for each packet, at each time slot. An algorithm parameter ϕ is used to control the number of iterations Value Iteration will update $J(s)$ for all states until all updates are smaller than ϕ. At that moment, it is considered that convergence occurred. The pseudo code for the algorithm used is described in Alg. 1

8.11 Numerical Results

In this section, numerical results obtained from solving the described problem are presented. The relevant simulation parameters used are presented in Tab. 13. The learned optimal policy is compared with three different benchmarks. First, a fixed policy where Lv4 is always chosen is considered. This policy represents the state of the art in most

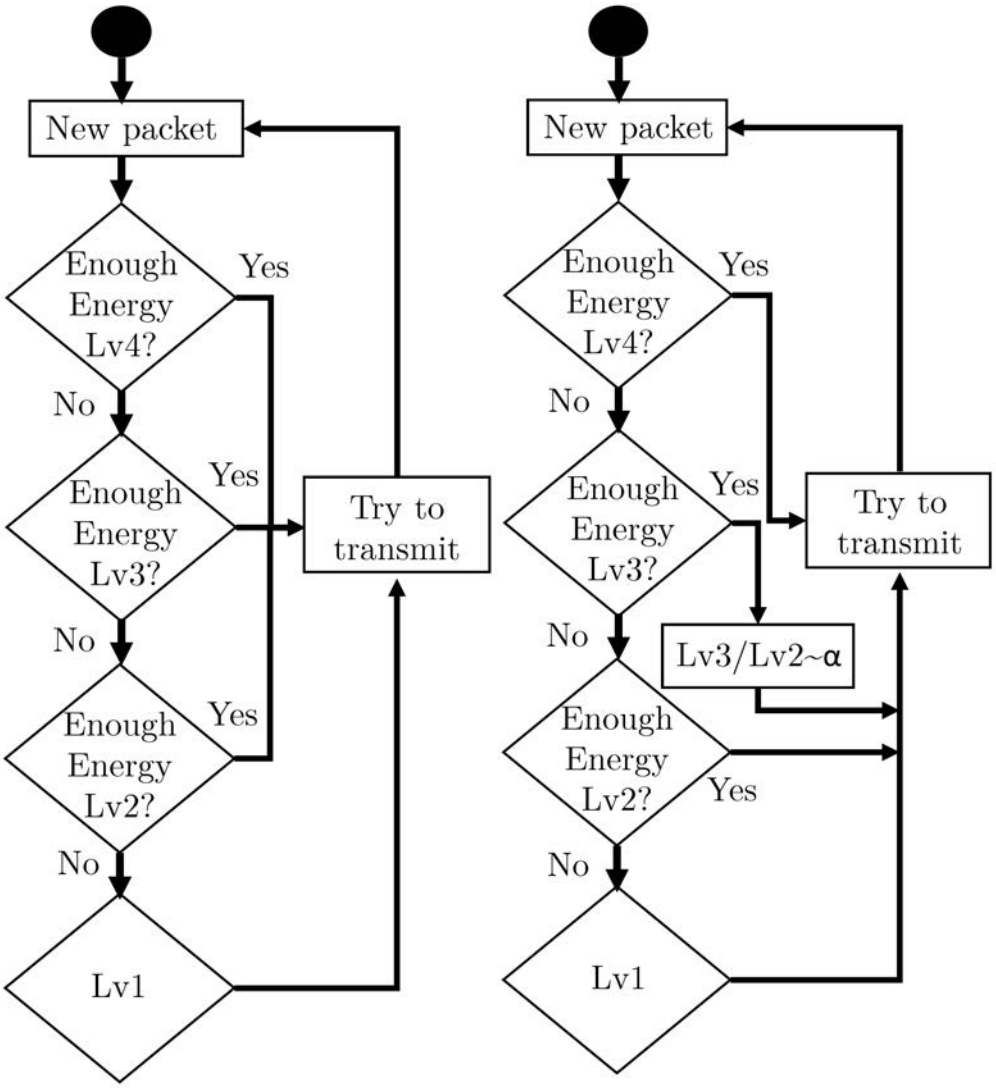
Algorithm 1 Value Iteration Algorithm

1: Initialisation:
set value update threshold $\phi = 0.001$
Randomly set $J(s) \in]0, 1], \forall s \in \mathcal{S}$
2: **while** $J(s)$ update $> \phi, \forall s \in \mathcal{S}$ **do**
For each state $s \in \mathcal{S}$ compute
$J(s) \leftarrow \max_{a_n \in \mathcal{A}_n} \sum_{d_n h_n} p_D p_H [F(c_n, b_n) + \gamma J(s_{n+1})]$
check if $J(s)$ update $< \phi$
3: **end while**
4: Output $\pi_* = \arg\max_{a_n \in \mathcal{A}_n} \sum_{d_n h_n} p_D p_H [F(c_n, b_n) + \gamma J(s_{n+1})]$

communicating systems. Then two other more elaborate policies were designed that is referred to as Smart Policy 1 (SP1) and Smart Policy 2 (SP2). Both these policies explore past knowledge the energy income and packet size. As in time slot $n + 1$, both h_{n+1} and d_{n+1} cannot be predicted, the policies use the last values of energy income and packet size, h_n and d_n, to verify if a packet can be transmitted. SP1 verifies if the highest security level can be attributed to the next packet and if not, lower levels are verified until the packet is discarded, if the predicted available energy is not sufficient not even for sending the packet unprotected, considering that $h_{n+1} = h_n$ and $d_{n+1} = d_n$. SP2 behaves similarly, but if it accesses that a packet can be sent with a Lv3 security, then it randomly selects Lv2 or Lv3 based on the security features parameter α. Flowcharts illustrating how SP1 and SP2 work are depicted in Figs. 22a and 22b.

There are three main goals in this section. First, the aim is to show how the choice of the tunable parameters α and β affects the authentication/confidentiality and energy/security tradeoffs. Tuning α dictates how authentication is preferred over confidentiality. Tuning β dictates how battery performance is more or less important than data security performance.

As these parameters tend to extremes on their possible values, predictable behavior occurs testing the resulting optimal policies. Low values of α cleverly result in a learned policy that behaves almost like a fixed policy on Lv2. If due to the designer's choice, confidentiality is the most important feature, then the learned policy avoids spending extra energy choosing Lv4 as this feature is provided with in Lv2 as well. A value $\alpha = 0$ reflects this behavior to perfection. In the opposite sense, a value $\alpha = 1$ tells the learning process that authentication is the only feature required. Therefore, the learned policy behaves like a fixed Lv3 policy, as soon as there is enough energy to do so. In the case of parameter β, lower values result in a tendency to ignore the battery durability. The extreme case of $\beta = 0$ will therefore result in a complete ramp up of the security level assigned and the battery performance will become neglected. On the other hand, $\beta = 1$ results in learning a policy that focus on transmitting packets without any feature assigned, i.e., in security Lv1, maximizing the battery performance and assuring that the biggest number of packets possible is transmitted. For these reasons, the extreme



(a) Smart policy 1

(b) Smart policy 2

Figure 22: Smart policies

cases from the results obtained are omitted.

Fig. 23 and Fig. 24 show results for how the tunable parameters α and β influence the average number of authenticated packets and the average available battery, respectively. Results in Fig. 23 are obtained by sweeping parameter α for different lower values of β, obtaining an optimal policy and simulating it in runtime. Results in Fig. 24 are obtained similarly, but sweeping this time β for different values of α.

As a second goal, the aim is at showing the benefits of learning optimal policies for this problem in terms of available energy in the battery and data reliability while still

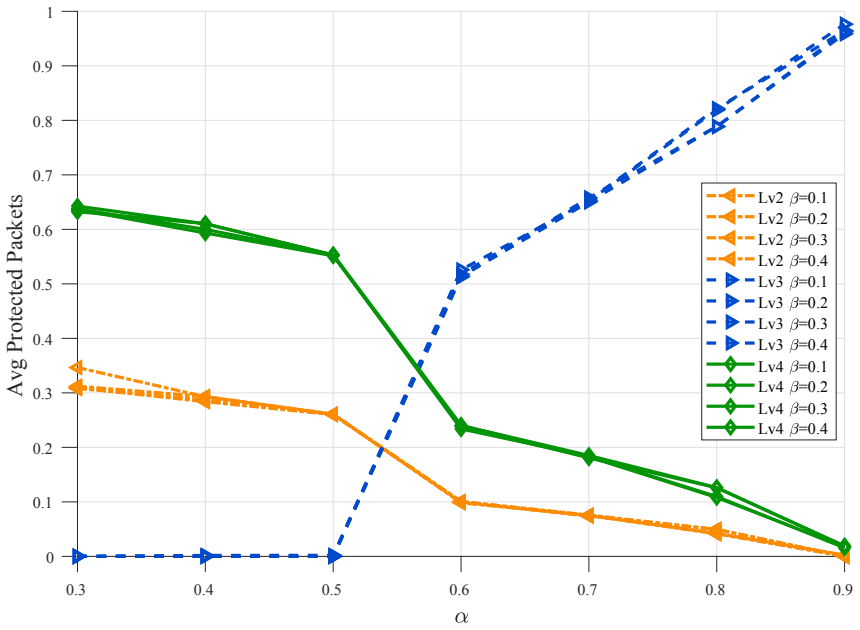


Figure 23: α control over Authenticated packets

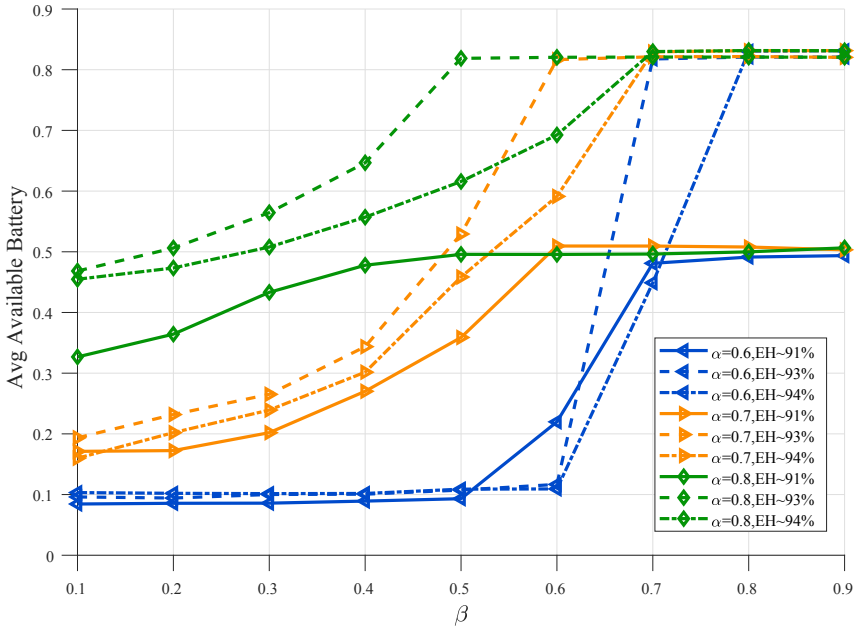


Figure 24: β control over Available battery

providing and maximising as much as possible data authentication and confidentiality protection, even though the energy income is not enough for normal, fully secure operation. The metrics considered are the average available battery and the number of discarded packets. Results obtained are illustrated in Fig. 25 and Fig. 26 and are obtained by arriving at the optimal policy for each value of the mean of H_n and comparing that with the benchmarks behavior for the same energy income, which also applies to Fig. 27, Fig. 28 and Fig. 29.

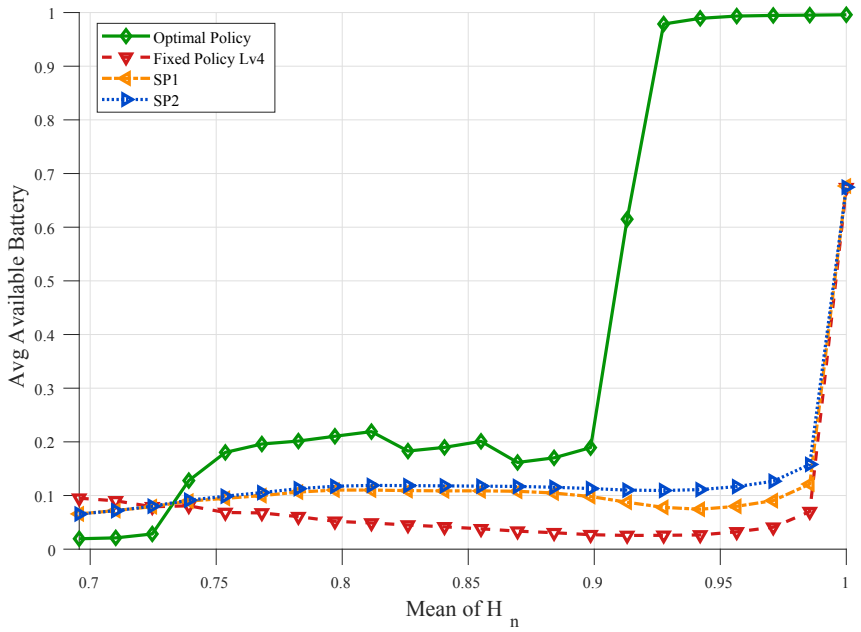


Figure 25: Average battery against energy income

The mean of H_n was normalised to the value that would be required to provide Lv4 protection to each packet, for easier reading of the benefits. The energy stored in the battery is normalised to b_{max}. The number of discarded packets is normalised to the total number of packets in the simulation (or the number of time slots as one packet per time slot is simulated). With around as much as 73% of the EH energy required for full protection, the energy available in the battery is already greater than any other policy. With around 90% of the same required EH energy, the energy performance is much superior than any other policy. At the same landmark values for the EH energy, the average discarded packets are very close to zero and zero.

The third goal is to show that packets can still be protected as much as possible, even with the very significant gains shown in Fig. 25 and Fig. 26. The security related metrics are the number authenticated packets, i.e., packets transmitted on Lv3 or Lv4, the number of packets protected for confidentiality, i.e., packets transmitted on Lv2 or Lv4 and finally, the number of packets sent with no security feature or with a security

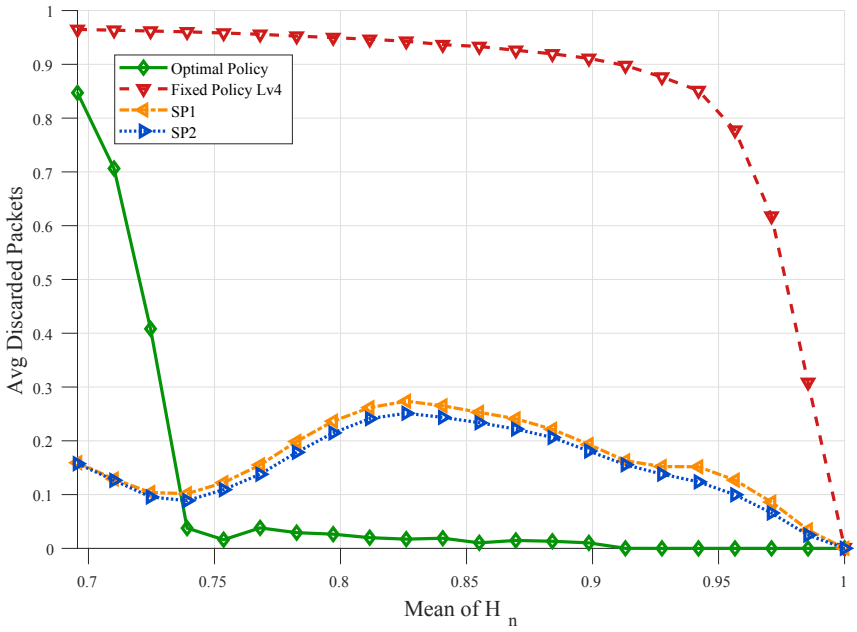


Figure 26: Average discarded packets against energy income

context Lv1. The results obtained for these metrics are illustrated respectively in Fig. 27, Fig. 28 and Fig. 29.

Table 13: Simulation parameters

Parameter	Value			
Simulation	Control	Offline	Online	Online Deep Learning (DL)
b_{max}	84	84	384	384
α	$\{0.6; 0.8\}$	0.8	0.8	0.8
β	$\{0.1; 0.4\}$	0.3	0.3	0.6
$H_n(\mu, \sigma)$	({63;65},1)	({43;69},1)	({43;69},1)	See plots
$D_n(\mu, \sigma)$	(40,1)	(40,1)	(40,1)	See plots
Avg rounds	5000			
Episodes	N/A		500	100
N	200			50
α_{TD}	0.1			
γ_{TD}	0.9			

Results show the number of authenticated packets by the optimal policy is far superior to all other policies, especially when compared to a fixed Lv4 policy. The optimal policy under performs in some regions in number of encrypted packets (protected for confidentiality) compared to other policies. This is not a concern as parameters have

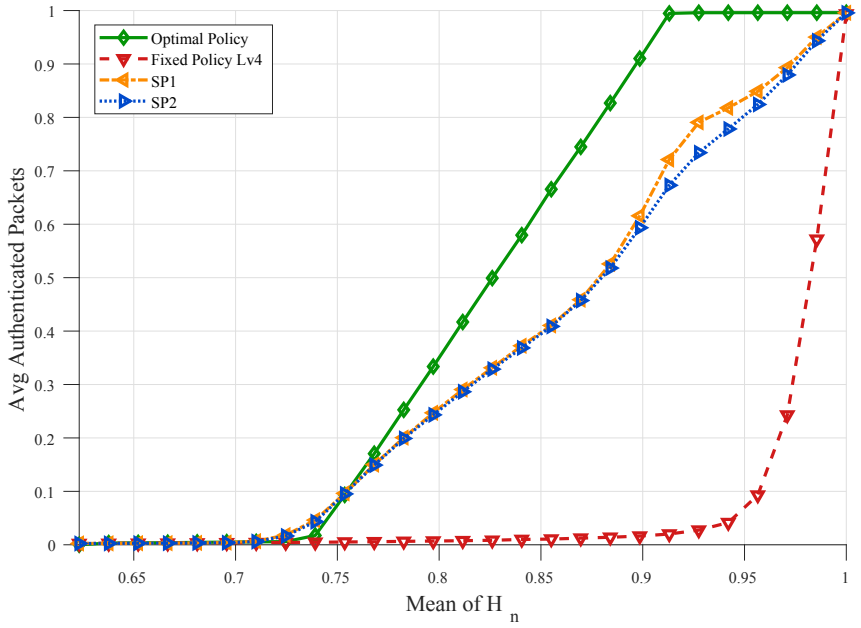


Figure 27: Authenticated packets against energy income

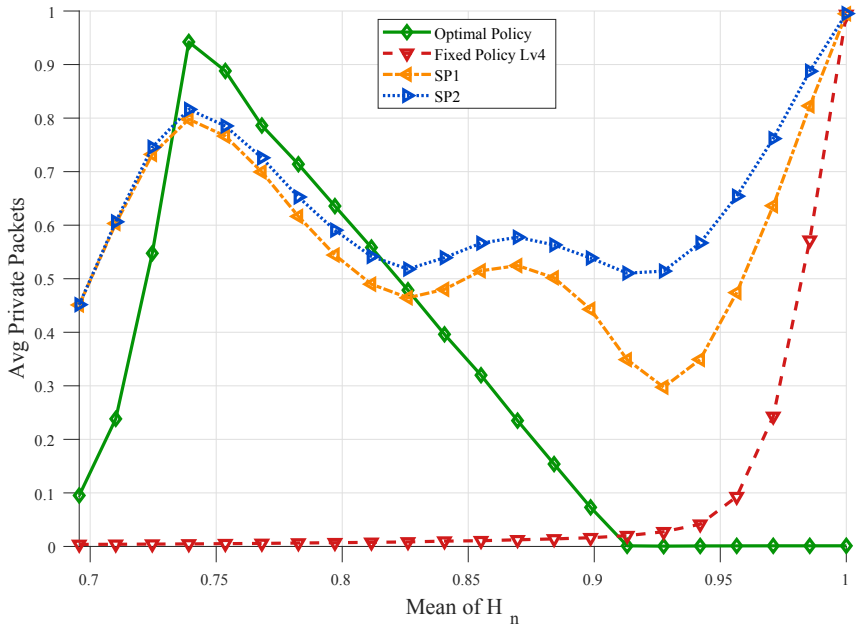


Figure 28: Confidential packets against energy income

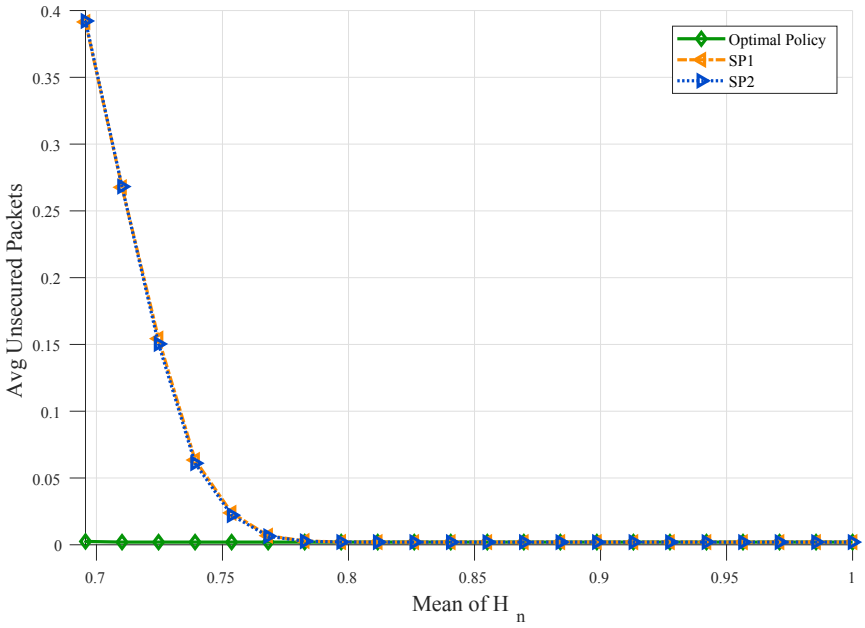


Figure 29: Packets sent unsecured against energy income

been set as $\alpha = 0.8$ and $\beta = 0.3$ and therefore emphasis is being given on data authentication and on battery available as well. It will be shown in Sec. 8.18 that the non optimal policies severely degrade their performance with changes in parameter β and that they never perform better considering all tradeoffs. This is also complemented with what is observed in Fig. 29, where the optimal policies protect all packets whereas SP1 and SP2 do neglect security.

8.12 Online learning

In the previous section, a completely modeled environment was described and its solution outlined via Value Iteration. The MDP was solved and then the benefits and gains achieved were shown using various different metrics. Despite all the gains, in this section there is a move from offline to consider online learning methods due to various reasons. First, dynamic programming methods require high computational effort [5], which can be unfeasible for resource constrained devices. Computing an optimal policy requires several iterations over the transition probabilities and reward matrices. The number of matrices grows linearly with the number of actions and their size grows exponentially with the number of states. This leads to severe scalability problems that prevent from simulating and obtaining optimal policies for large state spaces [5]. The scalability issue is therefore, directly linked with the computational complexity of finding the solution.

To tackle this problem, the benefits of online learning methods were studied. Several of these algorithms are available in the literature but there is no formal proof of perfor-

mance prediction in terms of computational complexity and resource requirements and especially, in terms of speed of convergence. In fact, it has be shown in the literature that online algorithms perform differently depending on the problem where they are applied [5]. Therefore a comparative study becomes necessary to attain which algorithms would perform better.

To achieve this goal, several TD learning methods have been implemented and applied to this problem. TD methods are on-line methods that learn through experience in real time. In these methods, there is a table of Q-values where the value of each state-action pair, i.e., the action value function $Q(s, a)$, is stored. The learning process is divided in episodes. An episode is composed of a finite sequence of time slots $n \in [0, N]$, where N is the length of the episode. An update to the Q-value corresponding to each visited state-action pair is made every time slot. This allows to perceive immediately that the learning process is much lighter in computational terms. Thus, an increase in the state space is affordable and more realistic models can be simulated and studied.

A downside to these approaches though is that the Q-values need to be stored. As the tendency is to study models with bigger state spaces, the memory requirements for these methods increase. For these reasons, a comparison between computational and memory resources needed for offline and online methods is required.

In the next sub sections, the algorithms applied in this work are briefly introduced and the used pseudo-code is depicted. These methods require a tradeoff between exploration and exploitation until convergence is reached. The learning process has the necessity of using random action selection to ensure all state-action pairs are visited a theoretically infinite number of times to assure convergence. A common way to provide exploration ability is by using ϵ-greedy action selection. An exploration parameter ϵ is thus used and actions are selected with a probability $1-\epsilon+\frac{\epsilon}{|\mathcal{A}_n(s)|}$ for the action with the highest value and a probability $\frac{\epsilon}{|\mathcal{A}_n(s)|}$ for all the others. A step parameter α_{TD} is also used to limit the weight of single state-action pair updates. This is commonly referred to as the learning rate. A discount-rate parameter γ_{TD} is also used. Immediate rewards are denoted as R. In the following sections, that update rule and complete algorithm pseudo-code are detailed.

8.13 SARSA

SARSA is an on-policy method, i.e., it follows a policy and uses that policy to update the Q-values on every time slot based on the pair s_{n+1}, a_{n+1} where the action for the next state is chosen based on the policy being followed. The Q-value update is thus defined as:

$$Q(s_n, a_n) = Q(s_n, a_n) + \alpha_{TD}[R + \gamma Q(s_{n+1}, a_{n+1}) - Q(s_n, a_n)] \tag{13}$$

A complete pseudo-code is given in the Alg. 2 box.

Algorithm 2 SARSA algorithm

Initialisation:
Set $\alpha_{TD} \in]0, 1]$
Set $\gamma_{TD} \in]0, 1]$
Randomly set $Q(s, a) \in]0, 1], \forall s \in \mathcal{S}, \forall a \in \mathcal{A}$
while Episode<number of Episodes **do**
$b_n \leftarrow b_{max}$ (start with a full battery)
$c_n \leftarrow Lv4$ (start with maximum security)
    while n<N **do**
Select random a_n and a_{n+1} based on ϵ
Take action a_n and transit to s_{n+1}
SARSA update - Eq. 13
$n \leftarrow n + 1$
    end while
end while
Output $\pi_*(s) = \underset{a_n \in \mathcal{A}_n s_n \in \mathcal{S}}{\arg\max} \{Q(s_n, a_n)\}, \forall s \in \mathcal{S}$

8.14 Expected SARSA

Expected SARSA is a variant from SARSA with a slightly different Q-value update rule. It that takes into account the expected value of the action in state s_{n+1}. The expected value is calculated based on the action selection probability, $P(a|s_{n+1})$, that in this work comes from an ϵ-greedy approach. Follows that the update rule is given by:

$$Q(s_n, a_n) = Q(s_n, a_n) + \alpha_{TD}[R + \gamma \sum_a P(a|s_{n+1})Q(s_{n+1}, a) - Q(s_n, a_n)]$$

A complete pseudo-code is given in the Alg. 3 box.

By making updates based on the expected value, the variance of those updates is reduced and thus, in many cases, Expected SARSA tends to perform better achieving faster convergence.

8.15 n-step SARSA

The first version of online learning through experience was coined Monte Carlo method [5]. These methods make updates to the Q-values at the end of each episode based on knowledge stored during an agent's interaction with the environment. Keeping record of all the state transitions, actions taken and rewards collected results in a big increase in memory requirements. Furthermore, the same state-action pair can be visited multiple times during the same episode which can easily result in a slow learning, due to the fact that the Q-value for that state-action pair is not immediately updated, which could result in choosing an under optimal action. For that reason, TD methods are widely regarded as faster learning methods [5]. In between these two ideas, n-step SARSA introduces a

Algorithm 3 Expected SARSA algorithm

1: Initialisation:
Set $\alpha_{TD} \in]0,1]$
Set $\gamma_{TD} \in]0,1]$
Randomly set $Q(s,a) \in]0,1], \forall s \in \mathcal{S}, \forall a \in \mathcal{A}$
2: **while** Episode<number of Episodes **do**
$b_n \leftarrow b_{max}$ (start with a full battery)
$c_n \leftarrow Lv4$ (start with maximum security)
3: **while** n<N **do**
Select random a_n based on ϵ
Take action a_n and transit to s_{n+1}
E-SARSA update - Eq. 14
$n \leftarrow n+1$
4: **end while**
5: **end while**
6: Output $\pi_*(s) = \underset{a_n \in \mathcal{A} n s_n \in \mathcal{S}}{\arg\max} \{Q(s_n, a_n)\}, \forall s \in \mathcal{S}$

step parameter for evaluating Q-value updates n time slots in the future. In this way, updates are not calculated every time slot. Instead, the agent stores information related to the experience, i.e., the states, actions and rewards observed during n time slots, and the update is calculated in future, delayed by the number of steps defined with n. This results in bigger memory requirements but it often shows faster learning results [5].

The Q-value update rule for n-step SARSA is given by:

$$\begin{aligned} G &\leftarrow \sum_{i=\tau+1}^{min(\tau+n,N)} \gamma^{i-\tau-1} R_i \\ Q(s_\tau) &\leftarrow Q(s_\tau) + \alpha_{TD}[G - Q(s_\tau)], \end{aligned} \tag{14}$$

where G is the return and τ is the current time slot. A complete pseudo-code is given in the Alg. 4 box.

8.16 Q-learning

The Q-learning algorithm also accounts for the immediate reward and the current state-action pair. However, its updates differ in which they find the action tha maximizes the value of the next state. The Q-learning update rule is given by:

$$Q(s_n, a_n) = Q(s_n, a_n) + \alpha_{TD}[R + \gamma \max_{a_{n+1}} Q(s_{n+1}, a_{n+1}) - Q(s_n, a_n)]$$

A complete pseudo-code is given in the Alg. 5 box.

Algorithm 4 n-Step SARSA algorithm

Initialisation:
Set $\alpha_{TD} \in]0,1]$
Set $\gamma_{TD} \in]0,1]$
Set $1 \leq n\text{-}step < N$
Randomly set $Q(s,a) \in]0,1], \forall s \in \mathcal{S}, \forall a \in \mathcal{A}$
while Episode<number of Episodes **do**
$b_n \leftarrow b_{max}$ (start with a full battery)
$c_n \leftarrow Lv4$ (start with maximum security)
Select random $a \sim \epsilon$-greedy policy
    while n<N **do**
Select random a_n and a_{n+1} based on ϵ
Take action a_n and transit to s_{n+1}
$\tau \leftarrow n - n\text{-}step + 1$

        if $\tau \geq 0$ **then** update G
            if $\tau + n\text{-}step < N$ **then**
$G \leftarrow G + \gamma^{n-step} Q(s_\tau, a_\tau)$
            end if
        end if
        make ϵ-greedy π wrt $Q(s,a)$
$n \leftarrow n + 1$
    end while
end while
Output $\pi_*(s) = \underset{a_n \in \mathcal{A}_n s_n \in \mathcal{S}}{\arg\max} \{Q(s_n, a_n)\}, \forall s \in \mathcal{S}$

8.17 Double Q-learning

Double Q-learning Its update rules:

$$Q_1(s,a) = Q_1(s,a) + \alpha_{TD}[R + \gamma Q_2(s\prime, \underset{a \in \mathcal{A}}{\arg\max}\, Q_1(s\prime, a) - Q(s,a)] \tag{15}$$

$$Q_2(s,a) = Q_2(s,a) + \alpha_{TD}[R + \gamma Q_1(s\prime, \underset{a \in \mathcal{A}}{\arg\max}\, Q_2(s\prime, a) - Q(s,a)] \tag{16}$$

A complete pseudo-code is given in the Alg. 6 box.

8.18 Online learning numerical results

In this section, numerical results are presented for the study of online TD methods. The relevant simulation parameters are presented in Tab. 13. The first analysis relates to n-step SARSA. There is no analytic method to prove what is the optimal value for the step parameter and therefore, a comparison between different values is required.

Algorithm 5 Q-learning algorithm

Initialisation:
small step parameter $\alpha_{TD} \in]0, 1]$
Randomly set $Q(s, a) \in]0, 1], \forall s \in \mathcal{S}, \forall a \in \mathcal{A}$
while Episode<number of Episodes **do**
$b_n \leftarrow b_{max}$
$c_n \leftarrow Lv4$
    while n<N **do**
Select random $a_n \sim \epsilon$-greedy policy
Take action a_n
$Q(s_n, a_n) \leftarrow Q(s_n, a_n) + \alpha_{TD}[R + \gamma \max_{a_{n+1} \in \mathcal{A}_{n+1}} Q(s_{n+1}, a_{n+1}) - Q(s_n, a_n)]$
Transit to s_{n+1}
$n \leftarrow n + 1$
    end while
end while
Output $\pi_* = \arg\max_{a_n \in \mathcal{A}_n s_n \in \mathcal{S}} \{Q(s_n, a_n)\}$

The best value is the one that allows the algorithm to achieve the highest reward value with the lowest experience time. Fig. 30 plots results from an incremental step increase, where the learning process is evaluated through its achieved reward after one episodic task, defined by $N = 200$ time slots.

It was found that the performance of different steps is similar in terms of achieving higher reward faster, although n-step values higher than three showed learning instability, i.e., the collected cumulative rewards while learning showed often a big decrease only to increase again a few episodes later. This unstable behavior can be seen as well for the results for 3-step SARSA. For visual simplicity reasons, Fig. 30 plots the best performing step values only and the step value of two is chosen as the step size that achieves the highest reward value with the lowest experience time while maintaining learning stability. A comparison is presented in Fig. 31 between 2-step SARSA, the remaining algorithms presented in the previous section, the previously described benchmarks, SP1, SP2 and a fixed policy for maximum security.

The results show very similar performance between all SARSA variants, slightly better performance for Q-learning and Double Q-learning by far outperforming all algorithms, achieving high reward levels much faster. Nevertheless, all the algorithms simulated outperform SP1 and SP2, whose cumulative reward values can be seen on the right axis in Fig. 31, and a Fixed Policy Lv4 that achieved an average cumulative reward $G = 0.05082$, that it was chosen not to plot due to visual considerations.

To ensure the Online RL approach completely outperforms the state of the art fixed policy, SP1 and SP2, a further comparison is needed evaluating their performance for all values of average energy income, μ from the Gaussian distribution H_n, that under feed the device energetically. This comparison is illustrated in Fig. 32.

Algorithm 6 Double Q-learning algorithm

Initialisation:
small step parameter $\alpha_{TD} \in]0, 1]$
Randomly set $Q_1(s, a), Q_2(s, a) \in]0, 1], \forall s \in \mathcal{S}, \forall a \in \mathcal{A}$
while Episode<number of Episodes **do**
$b \leftarrow b_{max}$
$c \leftarrow Lv4$
    while n<N **do**
Select random $a \sim \epsilon$-greedy policy wrt $Q_1 + Q_2$
Take action a
Update $Q_1(s, a)$ or $Q_2(s, a)$ with equal probabilty
Transit to s_{n+1}
$n \leftarrow n + 1$
    end while
end while
Output $\pi_* = \arg\max_{a \in \mathcal{A} s \in \mathcal{S}} \{Q_1(s, a) + Q_2(s, a)\}$

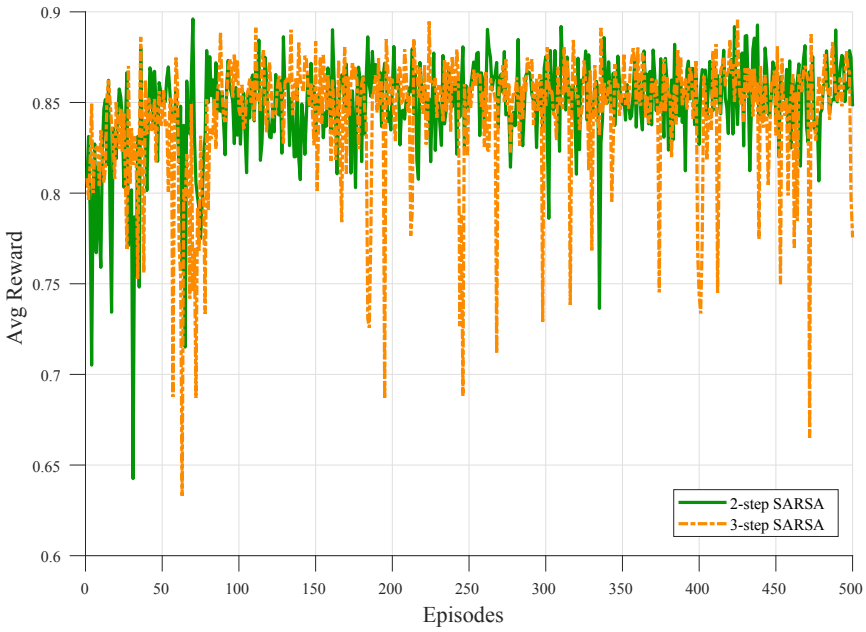


Figure 30: Comparison between different step values

Double Q-learning is used for the comparison as it was the best performing algorithm from Fig. 31. SP1, SP2 and a fixed policy Lv4 are all policies that aggressively try to maximise packet protection. Therefore, to attain a fair comparison, all four policies were tested for low values of β, where the optimal policies give more importance to security

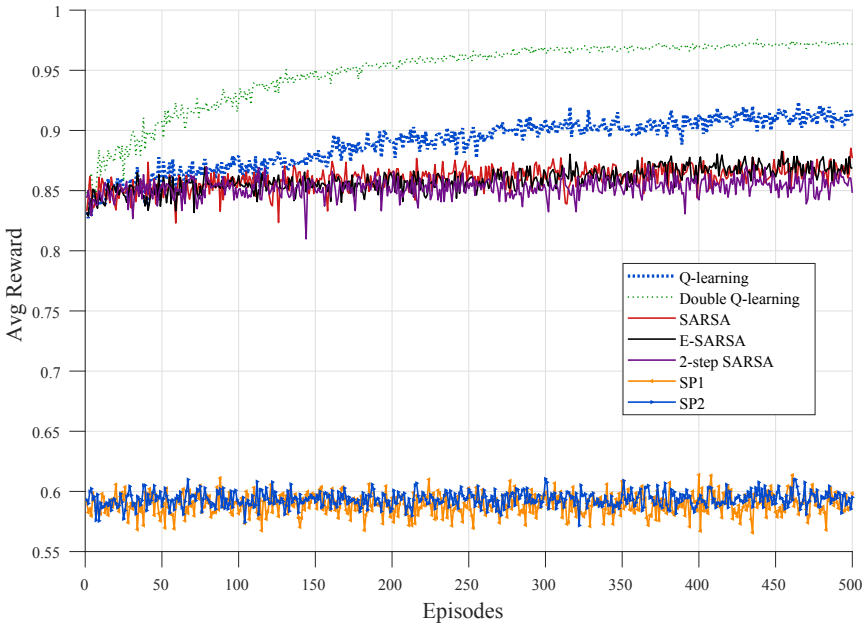


Figure 31: Average reward during learning

rather than energy. Results achieved show that the RL approach is not affected by variations of β values and completely outperforms the other policies that, as expected, suffer from a performance degradation as the concern for battery durability increases.

These implemented methods reduce significantly the computational effort but may still be a burden for devices with limited memory. This issue is addressed in the next section.

8.19 Online Deep Reinforcement learning

As previously discussed, online reinforcement learning methods have two major disadvantages. The need to store Q-values can be a burden in terms of memory, especially on resource constrained devices, and the speed of convergence to the optimal solution can be slow, causing that the learned policies on the training phase can be far from optimality which in turn can originate poor decision making. In a system model like the one presented in this work, the unpredictability of EH income and packet size pose the risk of poor decision making, leading to packet discarding, unsecured packet transmissions or lower energy efficiency.

To address these issues, an online deep reinforcement learning approach is presented in this section. To speed up the learning process, an actor-critic method [92] is used where both the policy and the action value function are parameterised with a ANN. This will also enable action value function approximation, eliminating the need for keeping Q-values stored in memory. The learning process is done by an online stochastic gradient

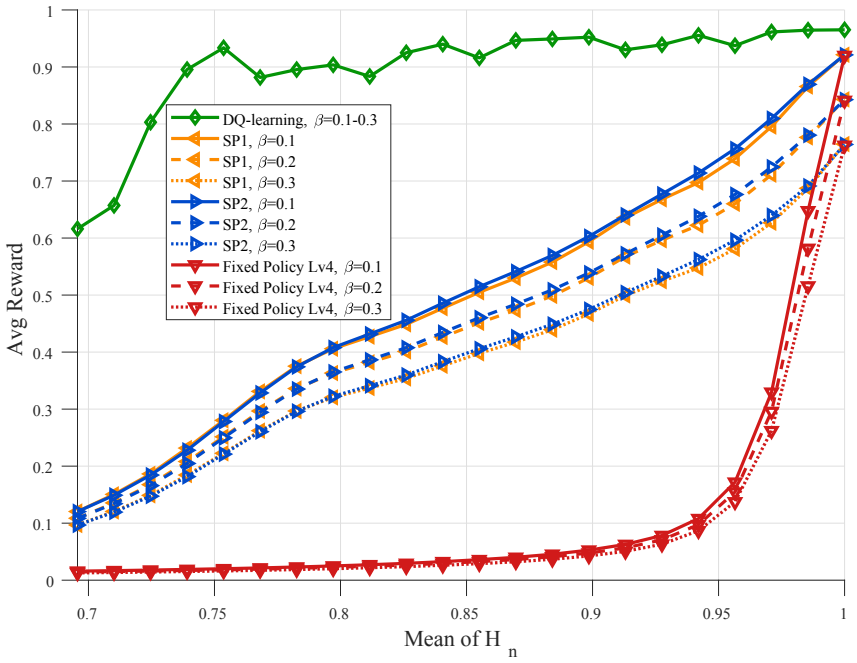


Figure 32: Cumulative Reward VS EH income

approach. The result is a stochastic policy where for each state s, a_n is selected based on a certain probability.

In the previous sections it was assumed a certain stability in terms of EH income and packet size, d_n. The target is now studying the behavior of the learned policies under extremely unstable conditions, where a stochastic policy is followed instead of a greedy, deterministic one.

8.20 Actor-Critic network

To approximate the action value function, a MLP with two layers as in Fig. 33 is considered. The input layer receives the state information, i.e., b_n and c_n and the output layer has one neuron for each possible action $a_n \in \mathcal{A}$. Each output neuron represents therefore the action value for the corresponding state-action pair, $Q^{\mathbf{w}}(s,a)$, where $\mathbf{w} \in \mathcal{R}^d$ is the vector that contains the weights that define the network and d is its dimensionality. Actor-critic methods usually require two ANN where one is used to parameterise the policy and the other is used to parameterise the action-value function. However in this work, because each action is being represented as an output neuron, it is considered that each of those neurons corresponds both to the approximated action-value, $Q^{\mathbf{w}}(s,a)$, and the approximated action preference, H_{a_n}. This can be seen as a network with two output layers, where one is used for the actor and the other for the critic. However the set of weights connecting the last hidden and output layers is the

same, getting updated depending on the output layer used for the training. Hence, only one ANN is required because it can approximate both the actor and the critic, an approach similar to the one used in [93].

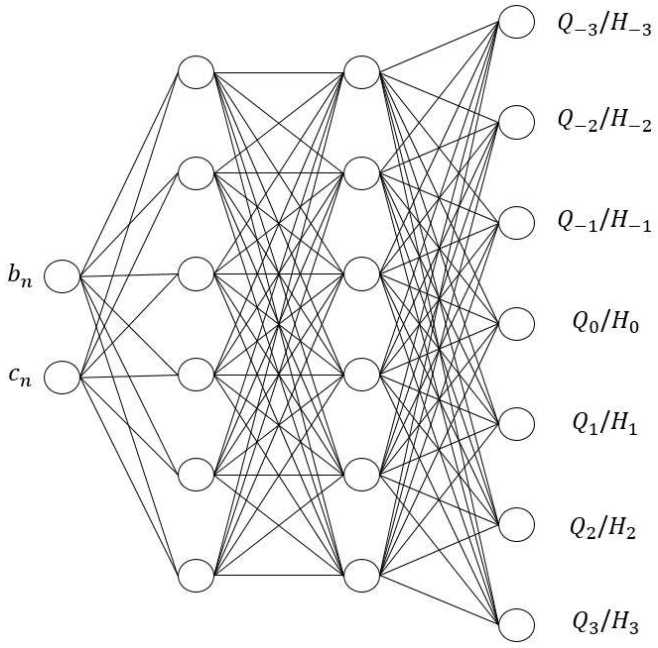


Figure 33: MLP model

The activation function used for the hidden layers is the TANH. This activation function has been used in several works and good training results have been shown using it. Although it is a popular activation function, some literature states that it may suffer from vanishing gradient problems. However in this work, it was found during training that it performs better when compared to other functions, namely the more recently widely used RELU and both in the hidden and output layers.

8.21 Training

To train the network, the total loss at the output layer needs to be calculated and back-propagated to the input layer, so that the parameterisation $\mathbf{w}$ can be adjusted. This procedure is executed every time slot and it is achieved by calculating the total loss, denoted as $\mathcal{L}$ and defined as the sum of the contributions of both the actor and critic output layer losses.

The critic loss is denoted as $\mathcal{L}_{Q^{\mathbf{w}}}$ and the Mean Squared Error (MSE) function is used for sampling the prediction error produced at each time slot n. The inputs for the MSE function are the predicted action value $Q^w(s_n, a_n)$ and the target value for time

slot n, denoted as t_n defined as

$$t_n = F(c_n, b_n) + \gamma Q^{\mathbf{w}}(s_{n+1}, a_{n+1}). \tag{17}$$

The target provides a measurement of how wrong the action value prediction was. However, an additional element is necessary to provide the gradient the direction needed on the update. The TD error at time slot n, denoted by δ_n, is therefore defined as

$$\delta_n = F(c_n, b_n) + \gamma Q^{\mathbf{w}}(s_{n+1}, a_{n+1}) - Q^{\mathbf{w}}(s_n, a_n). \tag{18}$$

The choice of the action a_{n+1} for state s_{n+1} is always greedy wrt to the action value function, considering only values of $Q^{\mathbf{w}}(s_{n+1}, a_{n+1})$ valid for $\mathcal{A}(s_{n+1})$. This makes this approach a Deep Q-Network (DQN) approach [94] and despite the fact it is used to show the next set of results, a more exploratory approach that randomly selects a_{n+1} according to the current action preferences derived from the current set of weights $\mathbf{w}$ was also tested, providing inferior results.

Combining all elements, the critic loss is obtained, defined as

$$\mathcal{L}_{Q^{\mathbf{w}}} = MSE(Q^{\mathbf{w}}(s, a), t_n) \times \delta_n \tag{19}$$

The actor loss contribution used is given by the policy gradient theorem [95], denoted as $\mathcal{L}_{\pi^{\mathbf{w}}}$ and defined as

$$\mathcal{L}_{\pi^{\mathbf{w}}} = -\log(\pi(a_n|s_n, \mathbf{w})) \times \delta_n. \tag{20}$$

Note that $-\log(\pi(a_n|s_n, \mathbf{w}))$ is evaluated only for the set of valid actions for each state, $\mathcal{A}(s)$. Finally, the total loss $\mathcal{L}$ is defined as

$$\mathcal{L} = \mathcal{L}_{Q^{\mathbf{w}}} + \mathcal{L}_{\pi^{\mathbf{w}}}. \tag{21}$$

For the optimization of the weights from the value of $\mathcal{L}$, the Adaptive Moment Estimation (ADAM) is used as it has been shown to be computationally more efficient than the other gradient descent variants [96]. Moreover, the learning rate is adapted to each weight in $\mathbf{w}$ during training, speeding up the convergence. Even if there is not enough training time, ADAM will converge faster to the best possible (lower) sample error, even if given limited training time [43].

A complete pseudo-code is given in the algorithm box.

8.22 Numerical results

In this section, numerical results obtained with the DQN approach are presented. The relevant simulation parameters are presented in Tab. 13. As previously mentioned, the purpose of implementing a Deep Learning approach is to address heavy memory requirements and confirm the expected better convergence properties [43]. Fig. 34 and Fig. 35 show the results obtained for the average reward and data availability. Plots were obtained by simulating the current policy after each episode and comparing the DQN approach with SP1 and SP2.

Algorithm 7 Actor-Critic algorithm

Initialisation:
small step parameter $\alpha_{TD} \in]0, 1]$
Randomly set $\mathbf{w}$
while Episode<number of Episodes **do**
Initialise s_0 randomly
    while n<N **do**
Sample and execute $a_n \sim \pi(a|s, \mathbf{w})$
Observe $s', F(c_n, b_n)$
Sample $a_{n+1} \sim$ greedy wrt $\pi(a|s, \mathbf{w})$
Compute eqs. (17) to (21)
Update $\mathbf{w}$
Transit to s_{n+1}
$n \leftarrow n + 1$
    end while
end while

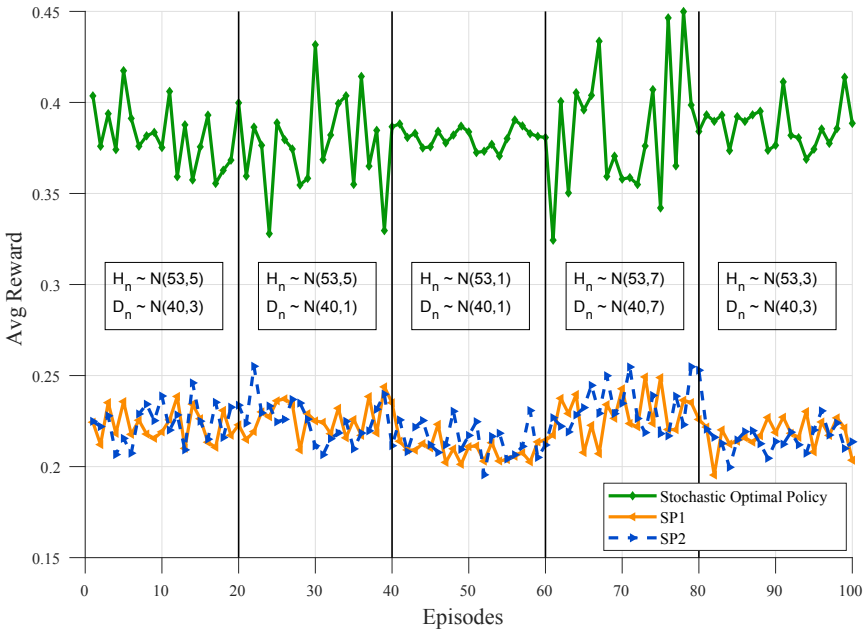


Figure 34: Learning stability analysis

Note the number of time slots per episode and the mean energy income were reduced compared to the ones used in the previous section. The purpose is twofold: 1) test the learning performance by reducing the learning time and by consequence, the number of updates of $\mathbf{w}$ and 2) degrade data availability in case choices for security level are poor.

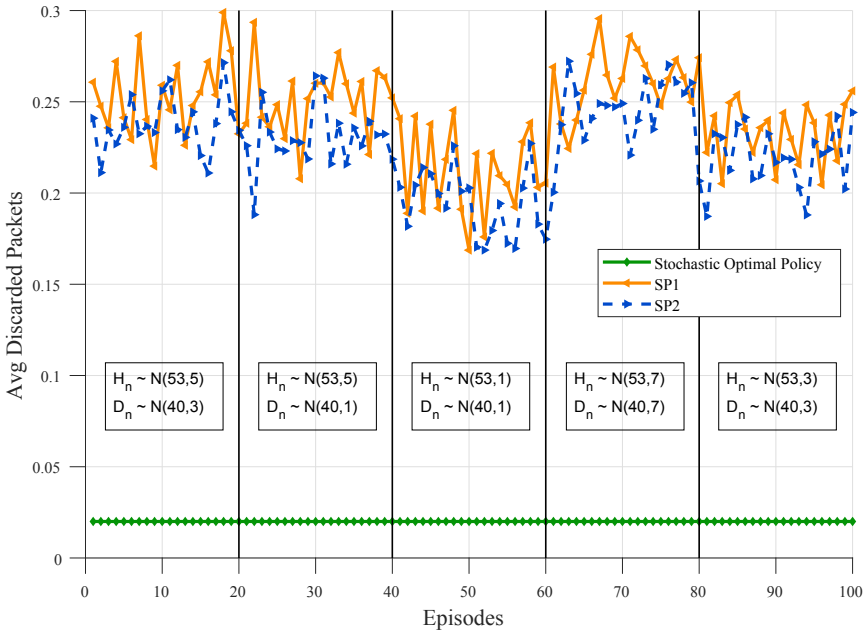


Figure 35: Data reliability stability

Every 20 episodes, the deviation for both H_n and D_n changed to bring instability to the sampled h_n and d_n. Results show the learned stochastic policies maintain good performance even under difficult learning conditions, achieving higher cumulative rewards and demonstrate a very high and stable level of data availability, indicating excellent security level decision making.

8.23 Memory requirements

To address the issue of the memory requirements, we now present their quantification for the different learning approaches used with eqs. (22) to (26).

$$M_{OFF} = 2|\mathcal{A}|(|c_n||b_{max}+1|)^2 \tag{22}$$

$$M_{ONL} = |c_n||b_{max}+1| \tag{23}$$

$$M_{DoubleQL} = 2|c_n||b_{max}+1| \tag{24}$$

$$M_{nstepSARSA} = |c_n||b_{max}+1| + 3\text{n-step} \tag{25}$$

$$M_{DL} = 2 + N_l \times N_n + |\mathcal{A}| \tag{26}$$

where M_{OFF} and M_{ONL} are the memory requirements for, respectively, the offline approach presented in Sec. 8.9, the Online RL methods described in Sec. 8.12 with the exception of two methods, Double Q-learning and n-step SARSA, to which the memory requirements are denoted by $M_{DoubleQL}$ and $M_{nstepSARSA}$, also respectively. Finally, M_{DL} denotes the requirements for the Online DL approach described in Sec. 8.19. The number of hidden layers and the number of neurons in each of these layers in the network defined by $\mathbf{w}$ is denoted by N_l and N_n.

Considering a 32 bit floating-point representation of the values involved and $b_{max} = 384$ for a fair comparison, the memory requirements are numerically quantified in Tab. 14. There is a clear descendant profile in terms of these requirements between the approaches tested, offline solved with Value Iteration (VI), online RL and online DL. Exception made to the offline approach, the memory footprint needed by the methods seems very reasonable considering the hardware of recent devices, even if resource constrained. However, considering $b_{max} = 84$ results in $M_{OFF} = 6,4736$ MB, which still is a reasonable value.

Table 14: Memory Requirements for all tested methods

Approach	Required Memory
M_{OFF}	132,8096 MB
M_{ONL}	6,160 KB
$M_{DoubleQL}$	12,320 KB
$M_{nstepSARSA}$	6,184 KB
M_{DL}	84 B

8.24 Remarks on the Optimization of the Security Context Selection

In this work, an approach for an security-energy tradeoff analysis has been presented. Devices under-fed by energy harvesters make decisions on the choice of security level to maximize protection of transmitted packets, data reliability and energy efficiency. Several approaches for offline and online learning were presented with significant gains achieved and the performance of the different methods was compared. Results from the DQN approach also show significant stability properties under difficult learning conditions followed by a study of the required memory requirements.

9 Conclusion

This dissertation addressed energetically sustainable 5G networks in futuristic smart cities scenarios, with a focus on security and its energy consumption. First, the smart cities scenarios were surveyed for an assessment of their main characteristics so that some security requirements could be learned, envisioned and formulated. It was concluded that smart cities are extremely connected to a city's inhabitants and all the benefits are projected to provide people with a better life quality and city experience. Mobility is regarded as one of the main aspects in smart cities, including all the movements people do on a daily basis. For that reason, the UE was almost immediately regarded as a tool to extend coverage and reduce connection distances in 5G towards the IoT.

The application of a dense IoT environment with the support of 5G networks is a mandatory component to the realization of this vision. It was found that this dense environment is estimated to have a very significant impact on the world's energy consumption. Following the studies related to energy generation and needs for future mobile and IoT networks, it is clear that the energy needs of the ICT ecosystems in general, and the 5G in particular will increase.

The world's electrical energy production still tends to rely immensely on sources of energy that are known to have extremely detrimental impact on the environment and contribute significantly for the carbon emissions. The available projections related to the global electrical energy production in the years to come point to a continuous and increased trend in relying on these detrimental sources such as coal. Dedicated EH equipment installed on the 5G network elements is a perfectly viable alternative solution to reduce the dependency of the energy generation on undesirable energy sources such as coal. It was however found that simply equipping 5G network elements with dedicated EH hardware is not enough to suffice the energy needs of end devices. Amongst others, the high demand in energy required for these networks to operate and the underdeveloped state of the latest small EH hardware solutions lead to the exploration of more energy efficient device cooperation strategies that in turn raise significant security concerns. These concerns are derived directly form the increased required interactions for the cooperation strategies to work.

The state of the art energy efficient device cooperation strategies were then surveyed and it was found that researchers focus on optimizing the selection of a routing path so that a packet flows from a source to a destination, achieving minimum energetic cost during that transmission. It was found that these approaches can be extended to consider communicating nodes with EH capabilities and that they do not consider the energy consumption due to the execution of security mechanisms, especially the cost of security establishment, that in some cases can be prohibitive.

From the study that was conducted to access the impact of security mechanisms on energy consumption, it was found that the available data in the literature has to be seen as indicative of the order of magnitude. And because singular opearations are quantified on all works surveyed, it was also found that this data is less useful for networking scenarios.

For these reasons, a new energy model for IoT devices has been introduced in Sec. 5. The model serves as a tool to quantify the energetic cost of establishing connections and of their secure communications phase, mapping these costs into different blocks commonly used in other energy models. It quantifies as well the cost of all modern cryptography algorithms. These algorithms are constantly being executed while a networking connection is established, active and then terminated. The results presented prove the model can be very useful in diverse networking scenarios, mainly by allowing to quantify the impact of single networking interactions or D2D connections to be clearly seen in a device's energy consumption. This visibility can be used in more complex IoT scenarios where techniques like load balancing or constraints like minimum thresholds for battery level have to be applied, due to the lack of visibility on the impact of single connections. This was presented by graphically identifying different energy efficiency zones when a device is advertising its availability to relay data for other network nodes. The presented results considered BLE security protocols and algorithms, demonstrating that it is not affordable for all devices, especially to constrained ones, to operate with the highest level of security described in this standard, although it is the recommended by the NIST.

Several D2D communication scenarios have also been layed out in Sec. 2.2. From the presented scenarios, some are already considered in the 5G and MTC, but some others are proposed in this dissertation. They relate to the direct connection between an UE and an IoT object. This proposal had as rationale the fact that modern smartphones and tablets can collect various types of information from their embedded sensors and they are also equipped with several connectivity options. Moreover, their mobility can enable connectivity in physical places where network infrastructure may be difficult or expensive to reach in terms of network coverage.

Based on this idea, a protocol for authentication and establishment of secure sessions between UEs and MTC devices without any prior trust was presented in Sec. 6. It relies on cryptographic systems that respect the nature of resource constrained MTC device and yet guarantee important security features. This proposal eliminates the need of MTC devices or Gateway (GW)s sending data to a server, saving significant amounts of energy, bandwidth and reducing latency. To the best of my knowledge, this is the first solution for this direct interaction in 5G and IoT. We introduce the ProSe standard to enhance the coverage of 5G by means of interaction between two UEs, or one UE and a GW and prove the security of the solution. This operating mode is therefore advocated since it is extremely efficient for coverage extension, saves energy and bandwidth and reduces communication distance. This solution is also very relevant and useful for PPDR scenarios, where UEs may play a key role in maintaining communications, and smart city scenarios where volatile type interactions often take place.

Security mechanisms have been addressed in this dissertation from the point of view of their functionality but mainly in relation to its energy consumption. This was due to the fact that the main concern in the SCAVENGE project is energy. The energetic consumption of security mechanisms and protocols has been surveyed and it was found that in the scientific literature, single executions on specific hardware are usually quantified. It was also found that the energetic consumption can vary significantly for the different

hardware platforms used for testing. This leads to the conclusion that the available values for energy consumption should be seen as indicative of the order of magnitude. This survey and the results and conclusions obtained in Sec. 5 also allowed to understand that the consumption due to the execution of security primitives may look residual when executed once, but have considerable impact in a medium to long term due to their continuous execution. The profiling of the energy consumption in UEs and IoT devices led to the conclusion that the transmissions are the main source of energy consumption in end devices and especially of IoT devices that do not have a screen, which also represents a significant consumption component in the case of UEs. After studying the security definitions for low power radios, namely BT, BLE and 802.15.4, it was found that the standards define security levels, based on the security establishment and the security features confidentiality, data integrity and data authentication, after security is established. These levels foresee already that some packets may not need full protection from these features. It was also found that protecting packets for confidentiality, data authentication and integrity requires appending extra Bytes of information to the protected packets so that it can be decrypted, and its data integrity and authenticity can be validated on the receiver end. Linked with the fact that transmissions are the main source of consumption on end devices, it became clear that security can have a significant impact on the total energy consumption of IoT and ProSe networks.

Therefore in Sec. 7, the concept of dynamic security levels based on security services for ProSe communications was introduced. Along with it, an inexpensive method for security context change in real time and a proposal for bootstrapping security parameters in UEs, totally compliant with ProSe standard and MIKEY protocol. The concept standardizes a common structure for all UEs, facilitates different IDS tasks and it allows for efficient energy saving and security increasing strategies. The context change method greatly reduces the security overhead and signaling in general between UEs, and completely to the CN. The bootstrapping of parameters mitigates the risk of downgrading attacks and defines the PF as the security policy responsible. The solution improves the visibility over malicious actions from the UE side and solves the problem of out of coverage UEs, that cannot get access to the CN for security information, enabling ProSe even in long time PPDR scenarios.

Finally, after realizing the impact that the security features of confidentiality, data integrity and data authentication have on the energy consumption while communicating, a need for a study of the security-energy tradeoff was required. The realization that both energy and security are fundamental characteristics in communications led to the study of this tradeoff. Particularly in this dissertation there is a major concern about the constrained resource nature of IoT devices energetically fed by EH hardware. The study of the state of the art small EH hardware to be suitable for mounting in also small IoT devices showed that the electrical energy generation is also a concern because of their limited energy generation capacity. If the EH hardware cannot produce enough energy for normal operation, it was proposed that the application of security features can be reduced as an effective power saving strategy. This is also backed up by the fact that low power radio standards already define some form of security levels, based on the idea

that not all packets require full protection. Recent developments in IDS also complement the idea that the energy spent on security features can be a considerable waste when no threat is present, which happens for most of the time in network operations, leading to a very low energy efficiency in cyber security.

The low energy efficiency and the strong impact of confidentiality and data authentication on the energy consumption of wireless devices lead to the proposal of an approach for an security-energy tradeoff analysis, presented in Sec. 8. A set of security levels was derived and a communications model was built where devices under-fed by energy harvesters make intelligent decisions on the choice of security level to maximize protection of transmitted packets, data reliability and energy efficiency. Three security features and energetic survival are tunable using two weight parameters. Several approaches for offline and online learning were presented to make the most suitable security level choice for each packet. Significant gains are achieved for available energy and data reliability while still providing security to packets. The proposed approach greatly increases the energy efficiency of the considered security features as they are used most of time without being needed. Several state of the art RL algorithms are compared, avoiding further unnecessary implementations to test their performance. Results obtained with DQN show very high stability properties under adverse learning conditions, showing it is an approach suitable for applications with unstable EH, often the case of, e.g., vibrational EH. A quantification of the memory footprint for all learning methods used was also presented, validating the idea that they are feasible approaches in modern, even if constrained devices.

10 Future Work

In this section, some future work ideas are presented. They are based upon the analysis of the proposed contributions with a focus on how they could be improved to continue the work developed in this dissertation.

In Sec. 5, an energy model was presented and some conclusions were derived from the simulations executed. It was concluded that mobility is an important aspect in smart cities scenarios, especially with the introduced idea of direct UE-IoT communications. The simulations presented in Sec. 5 showed that 1) mobility often introduces new nodes in a network and therefore, security needs to be established with them if they are to be part of routing path creation strategies. It was concluded that 2) establishing security using asymmetric cryptographic schemes can have a significant energetic cost and that 3) the networking load of an MTC device can greatly influence its ability to relay data for other devices. Due to these aspects, it was also concluded that load balancing techniques may not be the best strategy for routing path creation, especially if the network nodes are energetically fed by EH hardware.

For these reasons, the work could be extended exactly by a) developing routing path creation strategies that take the elements 1), 2) and 3) into account in order to further optimize existing works, b) to account for networks comprised of nodes fed by EH hardware and c) to eliminate the assumption made in the works surveyed that a trust relation already exists between all nodes in the considered network because as it was seen, the security establishment can be unaffordable for constrained devices, especially if executed often.

In the same work, it was also found that a relation exists between the cost of the active phase of a connection, E_{SC}, and its establishment, E_{CEM}. This relation could be further studied to find optimal packet buffering strategies based on security establishment methods. In applications were security establishment requirements are strict, the tradeoff between buffering time and the amount of data to be transmitted could be studied in order to optimize the number of concurrent transmissions. On the other hand, in applications were security establishment requirements are less strict, the tradeoff between the amount of data to be transmitted and the energy budget for security establishment could also be studied, with the same goal as to optimize the number of concurrent transmissions.

In Sec. 6, a protocol for authentication and establishment of secure sessions between UEs and MTC devices without any prior trust was presented. The end result is a direct, secure link between the two types of devices, duly authorized by the CN. This approach has proven to be more cost efficient than other works surveyed in the literature. This idea could be further extended to multiple MTC devices so that the UE could be capable of authenticating them, having as an end result direct connection to a group of MTC devices rather than just one.

In Sec. 7, the concept of dynamic security levels based on security services for ProSe communications was introduced. Along with it, a method for security context change in real time and a proposal for bootstrapping security parameters was also presented.

It was seen that two UEs communicating would still have to spend some extra energy on rekeying messages in order to change the current security level in use in real time. If EH capabilities were added to the UEs, an interesting optimization problem could be created in order to arrive at the best decision of security level and when to make that decision of changing it. This could serve to optimize the UEs energy usage taking into account their current battery and EH state, the cost of rekeying with a valid radio model and the cost of securing packets with the different available levels.

The defined security levels also could be extended to be more than four by including, e.g., the key size that serves as input to the security primitives in use at each level, similarly to what happens in the IEEE 802.15.4 standard. They could also be extended by including different security establishment methods (or no security establishment) as defined in the BT and BLE standards. If this proposal was extended in this way, then the bootstrapping of security parameters would also need to be changed so to account for the expansion of the security levels. Extending the security levels would be meaningful from a practical point of view and for completeness of the work, but it would probably have a smaller importance from a research perspective, if considered only by itself. However, if extended to be a part of the referred optimization problem, it would help to create an interesting and more complex problem.

In Sec. 8, a security-energy tradeoff analysis was presented. The set of security levels used was the same as in Sec. 7. Then, DP and RL approaches were applied to make intelligent decisions on the choice of security level. This approach could be extended to make the problem more complex and encompassing of needs and requirements that were found to be essential for IoT and ProSe networks.

The mentioned extension of the security levels could make the system model presented in Sec. 8 more complex. Especially in the case more security levels were used, it would make sense to define a constraint in the system model defined by a minimum security level that could be used.

The model defines two devices communicating where the security level decision is carried out by the transmitter node. This idea could be extended to a multipath routing where the residual energy of all potential participant nodes in the path would be considered, rather than just one.

Bibliography

[1] Mark P. Mills. The Cloud Begins with Coal. Technical report, August 2013.

[2] 3GPP. Proximity-based Services (ProSe); Security aspects (Release 14), June 2017.

[3] 3GPP. Service requirements for Machine-Type Communications, August 2017.

[4] Most powerful millimeter-scale energy harvester generates electricity from vibrations, April 2011.

[5] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, November 2018. Google-Books-ID: 6DKPtQEACAAJ.

[6] Jari Arkko, Elisabetta Carrara, Fredrik Lindholm, Karl Norrman, and Mats Naslund. MIKEY: Multimedia Internet KEYing.

[7] N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha. Analyzing the energy consumption of security protocols. In *Proceedings of the 2003 International Symposium on Low Power Electronics and Design, 2003. ISLPED '03.*, pages 30–35, August 2003.

[8] Dave Singelée, Stefaan Seys, Lejla Batina, and Ingrid Verbauwhede. The Energy Budget for Wireless Security: Extended Version. *IACR Cryptology ePrint Archive*, 2015:1029, 2015.

[9] A. Reziouk, E. Laurent, and J. Demay. Practical security overview of IEEE 802.15.4. In *2016 International Conference on Engineering MIS (ICEMIS)*, pages 1–9, September 2016.

[10] Kim Thuat Nguyen, Maryline Laurent, and Nouha Oualha. Survey on secure communication protocols for the Internet of Things. *Ad Hoc Networks*, 32:17–31, September 2015.

[11] Akhilesh Harsh and Nikhil Ichalkaranje. Transforming e-Government to Smart Government: A South Australian Perspective. In *Intelligent Computing, Communication and Devices*, Advances in Intelligent Systems and Computing, pages 9–16. Springer India, 2015.

[12] Debraj De. Sensor Networks for Smart Environments. *Journal of Telecommunications System & Management*, 3:1–2, February 2014.

[13] Hans Schaffers, Nicos Komninos, Marc Pallot, Brigitte Trousse, Michael Nilsson, and Alvaro Oliveira. Smart cities and the future internet: towards cooperation frameworks for open innovation. In *The Future Internet*, pages 431–446. Springer-Verlag, Berlin, Heidelberg, 2011.

[14] R. R. Harmon, E. G. Castro-Leon, and S. Bhide. Smart cities and the Internet of Things. In *2015 Portland International Conference on Management of Engineering and Technology (PICMET)*, pages 485–494, August 2015.

[15] Julius Sechang Mboli. Feasibility Study on Disaster Management with Hybrid Network of LTE and Satellite Links. *The Computing Research Repository*, abs/1609.02375, September 2016. arXiv: 1609.02375.

[16] Arun Mahizhnan. Smart cities: The Singapore case. *Cities*, 16(1):13–18, February 1999.

[17] Nancy Odendaal. Information and communication technology and local governance: understanding the difference between cities in developed and emerging economies. *Computers, Environment and Urban Systems*, 27(6):585–607, November 2003.

[18] Lynn M. McAllister, Helen M. Hall, Helen L. Partridge, and Gillian C. Hallam. Effecting social change in the 'smart city': the West End connect community project. In Chanel Bailey and Karen Barnett, editors, *Social Change in the 21 st Century*, pages 1–16, Brisbane, Australia, 2005.

[19] Chuantao Yin, Zhang Xiong, Hui Chen, JingYuan Wang, Daven Cooper, and Bertrand T. David. A literature survey on smart cities. *Science China Information Sciences*, 58(10), October 2015.

[20] Rudolf Giffinger, Christian Fertner, Hans Kramar, Evert Meijers, Dipling Christian Fertner, Dipling Dr, and Hans Kramar. *City-ranking of European Medium-Sized Cities*. Vienna University of Technology, 2007.

[21] Jungwoo Lee and Hyejung Lee. Developing and validating a citizen-centric typology for smart city services. *Government Information Quarterly*, 31:S93–S105, June 2014.

[22] Alejandro Moreno-Gomez, Carlos A. Perez-Ramirez, Aurelio Dominguez-Gonzalez, Martin Valtierra-Rodriguez, Omar Chavez-Alegria, and Juan P. Amezquita-Sanchez. Sensors Used in Structural Health Monitoring. *Archives of Computational Methods in Engineering*, 25(4):901–918, November 2018.

[23] M. Vučinić, B. Tourancheau, F. Rousseau, A. Duda, L. Damon, and R. Guizzetti. Energy cost of security in an energy-harvested IEEE 802.15.4 Wireless Sensor Network. In *2014 3rd Mediterranean Conference on Embedded Computing (MECO)*, pages 198–201, June 2014.

[24] X. Wang, J. Zhang, E. M. Schooler, and M. Ion. Performance evaluation of Attribute-Based Encryption: Toward data privacy in the IoT. In *2014 IEEE International Conference on Communications (ICC)*, pages 725–730, June 2014.

[25] P. Trakadas, T. Zahariadis, H. C. Leligou, S. Voliotis, and K. Papadopoulos. Analyzing energy and time overhead of security mechanisms in Wireless Sensor Networks. In *15th International Conference on Systems, Signals and Image Processing*, pages 137–140, June 2008.

[26] Christopher Huth, René Guillaume, Paul Duplys, Kumaragurubaran Velmurugan, and Tim Güneysu. On the Energy Cost of Channel Based Key Agreement. In *Proceedings of the 6th International Workshop on Trustworthy Embedded Devices*, TrustED '16, pages 31–41, Vienna, Austria, 2016.

[27] X. Zhang, H. M. Heys, and C. Li. Energy cost of cryptographic session key establishment in a wireless sensor network. In *6th International ICST Conference on Communications and Networking in China (CHINACOM)*, pages 335–339, August 2011.

[28] Panagiotis Ilia, George Oikonomou, and Theo Tryfonas. Cryptographic Key Exchange in IPv6-Based Low Power, Lossy Networks. In David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, Gerhard Weikum, Lorenzo Cavallaro, and Dieter Gollmann, editors, *Information Security Theory and Practice. Security of Mobile and Cyber-Physical Systems*, volume 7886, pages 34–49. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[29] G. de Meulenaer, F. Gosset, F. Standaert, and O. Pereira. On the Energy Cost of Communication and Cryptography in Wireless Sensor Networks. In *2008 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, pages 580–585, October 2008.

[30] 3GPP. System improvements for Machine-Type Communications, September 2012.

[31] 3GPP. Proximity-based services (ProSe), June 2017.

[32] Malka N. Halgamuge, Moshe Zukerman, Kotagiri Ramamohanarao, and Hai L. Vu. An Estimation of Sensor Energy Consumption. *Progress In Electromagnetics Research*, 12:259–295, 2009.

[33] Hai-Ying Zhou, Dan-Yan Luo, Yan Gao, and De-Cheng Zuo. Modeling of Node Energy Consumption for Wireless Sensor Networks. *Wireless Sensor Network*, 3(1):720–726, January 2011.

[34] John Padgette, John Bahr, Mayank Batra, Marcel Holtmann, Rhonda Smithbey, Lidong Chen, and Karen Scarfone. Guide to Bluetooth Security. *Special Publication (NIST SP) - 800-121 Rev 2*, May 2017.

[35] SIG Working Group. Bluetooth Specifications, 2019.

[36] W. K. G. Seah, Z. A. Eu, and H. Tan. Wireless sensor networks powered by ambient energy harvesting (WSN-HEAP) - Survey and challenges. In *2009 1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace Electronic Systems Technology*, pages 1–5, May 2009.

[37] Mohamed Hamid Elsheikh, Dhafer Abdulameer Shnawah, Mohd Faizul Mohd Sabri, Suhana Binti Mohd Said, Masjuki Haji Hassan, Mohamed Bashir Ali Bashir, and Mahazani Mohamad. A review on thermoelectric renewable energy: Principle parameters that affect their performance. *Renewable and Sustainable Energy Reviews*, 30:337–355, February 2014.

[38] [PDF] Design and analysis of a thermoelectric energy harvesting system for powering sensing nodes in nuclear power plant - Semantic Scholar.

[39] Sushanta Kundu and Harshal B. Nemade. Modeling and Simulation of a Piezoelectric Vibration Energy Harvester. *Procedia Engineering*, 144:568–575, January 2016.

[40] A. Erturk and D. J. Inman. Broadband piezoelectric power generation on high-energy orbits of the bistable Duffing oscillator with electromechanical coupling. *Journal of Sound and Vibration*, 330(10):2339–2353, May 2011.

[41] C. R. Valenta and G. D. Durgin. Harvesting Wireless Power: Survey of Energy-Harvester Conversion Efficiency in Far-Field, Wireless Power Transfer Systems. *IEEE Microwave Magazine*, 15(4):108–120, June 2014.

[42] Ari Keranen, Mehmet Ersue, and Carsten Bormann. Terminology for Constrained-Node Networks.

[43] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[44] Filipe Conceição, Nouha Oualha, and Djamal Zeghlache. An Energy Model for the IoT: Secure Networking Perspective. September 2018.

[45] Q. Wang, M. Hempstead, and W. Yang. A Realistic Power Consumption Model for Wireless Sensor Network Devices. In *2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, volume 1, pages 286–295, September 2006.

[46] Modeling Energy Consumption of Wireless Sensor Networks by SystemC - IEEE Conference Publication.

[47] G. Steri, G. Baldini, I. N. Fovino, R. Neisse, and L. Goratti. A novel multi-hop secure LTE-D2d communication protocol for IoT scenarios. In *2016 23rd International Conference on Telecommunications (ICT)*, pages 1–6, May 2016.

[48] A. Ometov, K. Zhidanov, S. Bezzateev, R. Florea, S. Andreev, and Y. Koucheryavy. Securing Network-Assisted Direct Communication: The Case of Unreliable Cellular Connectivity. In *2015 IEEE Trustcom/BigDataSE/ISPA*, volume 1, pages 826–833, August 2015.

[49] Luca Caviglione, Alessio Merlo, and Mauro Migliardi. Green-Aware Security: Towards a new Research Field. page 10.

[50] L. Lazos and R. Poovendran. Energy-aware secure multicast communication in ad-hoc networks using geographic location information. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, volume 4, pages IV–201, April 2003.

[51] Young-Pil Kim, Seehwan Yoo, and Chuck Yoo. DAoT: Dynamic and energy-aware authentication for smart home appliances in Internet of Things. In *2015 IEEE International Conference on Consumer Electronics (ICCE)*, pages 196–197, January 2015.

[52] Thaier Hayajneh, Razvi Doomun, Ghada Al-Mashaqbeh, and Bassam J. Mohd. An energy-efficient and security aware route selection protocol for wireless sensor networks. *Security and Communication Networks*, 7(11):2015–2038, 2014.

[53] Nidal Nasser and Yunfeng Chen. SEEM: Secure and energy-efficient multipath routing protocol for wireless sensor networks. *Computer Communications*, 30(11):2401–2412, September 2007.

[54] Energy Consumption of Cryptographic Algorithms in Mobile Devices.

[55] Arcangelo Castiglione, Francesco Palmieri, Ugo Fiore, Aniello Castiglione, and Alfredo De Santis. Modeling energy-efficient secure communications in multi-mode wireless mobile devices. *Journal of Computer and System Sciences*, 81(8):1464–1478, December 2015.

[56] A. Castiglione, A. D. Santis, A. Castiglione, F. Palmieri, and U. Fiore. An Energy-Aware Framework for Reliable and Secure End-to-End Ubiquitous Data Communications. In *2013 5th International Conference on Intelligent Networking and Collaborative Systems*, pages 157–165, September 2013.

[57] J. Duan, D. Gao, D. Yang, C. H. Foh, and H. Chen. An Energy-Aware Trust Derivation Scheme With Game Theoretic Approach in Wireless Sensor Networks for IoT Applications. *IEEE Internet of Things Journal*, 1(1):58–69, February 2014.

[58] A. Ahmed, K. A. Bakar, M. I. Channa, K. Haseeb, and A. W. Khan. TERP: A Trust and Energy Aware Routing Protocol for Wireless Sensor Network. *IEEE Sensors Journal*, 15(12):6962–6972, December 2015.

[59] Energy-efficient mechanisms in security of the internet of things: A survey | Elsevier Enhanced Reader.

[60] Opportunistic Routing Algorithm for Relay Node Selection in Wireless Sensor Networks - IEEE Journals & Magazine.

[61] Kassio Machado, Denis Rosário, Eduardo Cerqueira, Antonio A. F. Loureiro, Augusto Neto, and José Neuman de Souza. A Routing Protocol Based on Energy and Link Quality for Internet of Things Applications. *Sensors (Basel, Switzerland)*, 13(2):1942–1964, February 2013.

[62] SCAVENGE-Sustainable Cellular Network Harvesting Ambient Energy. WP2–Energy Models, 2018.

[63] B. Martinez, M. Montón, I. Vilajosana, and J. D. Prades. The Power of Models: Modeling Power Consumption for IoT Devices. *IEEE Sensors Journal*, 15(10):5777–5789, October 2015.

[64] W. Du, F. Mieyeville, and D. Navarro. Modeling Energy Consumption of Wireless Sensor Networks by SystemC. In *2010 Fifth International Conference on Systems and Networks Communications*, pages 94–98, August 2010.

[65] F. Conceicao, N. Oualha, and D. Zeghlache. Security establishment for IoT environments in 5g: Direct MTC-UE communications. In *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 1–5, October 2017.

[66] N. Nikaein, M. Laner, K. Zhou, P. Svoboda, D. Drajic, M. Popovic, and S. Krco. Simple Traffic Modeling Framework for Machine Type Communication. In *ISWCS 2013; The Tenth International Symposium on Wireless Communication Systems*, pages 1–5, August 2013.

[67] N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha. A study of the energy consumption characteristics of cryptographic algorithms and security protocols. *IEEE Transactions on Mobile Computing*, 5(2):128–143, February 2006.

[68] Myungsik Yoo, Bui Xuan Yen, and Do Trong Hop. Redundant Transmission in Wireless Networked Control System over IEEE 802.15.4e. In *Proceedings of the 2013 International Conference on Information Networking (ICOIN)*, ICOIN '13, pages 628–631, Washington, DC, USA, 2013. IEEE Computer Society.

[69] Cooperative effort based wireless sensor network clustering algorithm for smart home application - IEEE Conference Publication.

[70] S. Yu, X. Wu, P. Wu, D. Wu, H. Dai, and G. Chen. CIRF: Constructive interference-based reliable flooding in asynchronous duty-cycle wireless sensor networks. In *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 2734–2738, April 2014.

[71] Peter Jonsson and Stephen Carson. Ericsson Mobility Report. SectionStartPage, June 2019.

[72] Andrea Goldsmith. *Cellular Systems and Infrastructure-Based Wireless Networks*. August 2005.

[73] 3GPP. Security aspects of Machine-Type Communications (MTC) and other mobile data applications communications enhancements, December 2016.

[74] National Institute of Standards and Technology. The Keyed-Hash Message Authentication Code (HMAC). Technical Report NIST FIPS 198-1, National Institute of Standards and Technology, Gaithersburg, MD, July 2008.

[75] M J Dworkin. Recommendation for block cipher modes of operation: the CMAC mode for authentication. Technical Report NIST SP 800-38b, National Institute of Standards and Technology, Gaithersburg, MD, 2016.

[76] L Chen. Recommendation for key derivation using pseudorandom functions (revised). Technical Report NIST SP 800-108, National Institute of Standards and Technology, Gaithersburg, MD, 2009.

[77] Bruno Blanchet. Automatic verification of correspondences for security protocols*. *Journal of Computer Security*, 17(4):363–434, July 2009.

[78] C. Lai, H. Li, R. Lu, Rong Jiang, and X. Shen. LGTH: A lightweight group authentication protocol for machine-type communication in LTE networks. In *2013 IEEE Global Communications Conference (GLOBECOM)*, pages 832–837, December 2013.

[79] 3GPP. Proximity-based Services (ProSe) - Security aspects.

[80] L. Caviglione, M. Gaggero, E. Cambiaso, and M. Aiello. Measuring the Energy Consumption of Cyber Security. *IEEE Communications Magazine*, 55(7):58–63, July 2017.

[81] J. Toldinas, R. Damasevicius, A. Venckauskas, T. Blazauskas, and J. Ceponis. Energy Consumption of Cryptographic Algorithms in Mobile Devices. *Elektronika ir Elektrotechnika*, 20(5):158–161, May 2014.

[82] ICT Facts and Figures 2017. Technical report, ICT, 2017.

[83] Shihua Cao and Jianqing Li. A survey on ambient energy sources and harvesting methods for structural health monitoring applications. *Advances in Mechanical Engineering*, 9(4):168781401769621, April 2017.

[84] Chongfeng Wei and Xingjian Jing. A comprehensive review on vibration energy harvesting: Modelling and realization. *Renewable and Sustainable Energy Reviews*, 74(C):1–18, 2017.

[85] How Long is the Lifetime of a Wireless Sensor Network? - IEEE Conference Publication.

[86] Zigbee Alliance. ZigBee Specification, September 2012.

[87] IEEE 802.15.4-2015 - IEEE Standard for Low-Rate Wireless Networks, 2015.

[88] Aleksandar Milenkoski, Marco Vieira, Samuel Kounev, Alberto Avritzer, and Bryan D. Payne. Evaluating Computer Intrusion Detection Systems: A Survey of Common Practices. *ACM Comput. Surv.*, 48(1):12:1–12:41, September 2015.

[89] Mohiuddin Ahmed, Abdun Naser Mahmood, and Jiankun Hu. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60:19–31, January 2016.

[90] Morris Dworkin. Recommendation for Block Cipher Modes of Operation: the CCM Mode for Authentication and Confidentiality. Technical Report NIST Special Publication (SP) 800-38C, National Institute of Standards and Technology, July 2007.

[91] Morris Dworkin. Recommendation for Block Cipher Modes of Operation: Methods and Techniques. Technical Report NIST Special Publication (SP) 800-38A, National Institute of Standards and Technology, December 2001.

[92] Vijay R Konda and John N Tsitsiklis. Actor-Critic Algorithms. page 7.

[93] Mastering the game of Go without human knowledge | Nature.

[94] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with Deep Reinforcement Learning. *arXiv:1312.5602 [cs]*, December 2013. arXiv: 1312.5602.

[95] Richard S. Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In *Proceedings of the 12th International Conference on Neural Information Processing Systems*, NIPS'99, pages 1057–1063, Cambridge, MA, USA, 1999. MIT Press. event-place: Denver, CO.

[96] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, December 2014. arXiv: 1412.6980.