Edge-assisted Task Offloading in the Internet of Wearable Things

Waleed Bin Qaim

Tampere University

Supervisors: Prof. Jari Nurmi (TAU)
Prof. Elena Simona Lohan (TAU)
Prof. Antonella Molinaro (UNIRC)
Prof. Claudia Campolo (UNIRC)
Dr. Aleksandr Ometov (TAU)
Ilaria Lener (T6 EcoSystems)

This thesis has been completed in a double Doctoral Degree programme at Tampere University (TAU), Finland and University Mediterranea of Reggio Calabria (UNIRC), Italy.

Tampere (Finland) March 2024

THESIS UNDER JOINT SUPERVISION

TAMPERE UNIVERSITY

UNIVERSITA' MEDITERRANEA DI REGGIO CALABRIA

DOCTORAL PROGRAM IN DYNAMIC WEARABLE APPLICATIONS WITH PRIVACY CONSTRAINTS

FACULTY OF INFORMATION TECHNOLOGY AND COMMUNICATION SCIENCES

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE, DELLE INFRASTRUTTURE E DELL'ENERGIA SOSTENIBILE (DIIES)

PHD IN INFORMATION ENGINEERING ciclo XXXV

EDGE-ASSISTED TASK OFFLOADING IN THE INTERNET OF WEARABLE THINGS

CANDIDATE

WALEED BIN QAIM

SUPERVISORS

Prof. Jari Nurmi (Tampere University)

Prof. Elena Simona Lohan (Tampere University)

Prof. Antonella Molinaro (University Mediterranea of Reggio Calabria)

Prof. Claudia Campolo (University Mediterranea of Reggio Calabria)

Dr. Aleksandr Ometov (Tampere University)

Ilaria Lener (T6 EcoSystems)

Tampere, March 2024

WALEED BIN QAIM

Edge-assisted Task Offloading in the Internet of Wearable Things

ACADEMIC DISSERTATION
To be presented, with permission of the
Faculty of Information Technology and Communication Sciences
of Tampere University and the Doctoral School of
University Mediterranea of Reggio Calabria,
for public discussion in the auditorium S4
of the Sähkötalo, Korkeakoulunkatu 3, Tampere,
on 16 April 2024, at 12 o’clock.

ACADEMIC DISSERTATION

Tampere University,
Faculty of Information Technology and Communication Sciences, Finland

University Mediterranea of Reggio Calabria,
Doctoral School, Italy

Responsible supervisor and Custos	Prof. Jari Nurmi Tampere University, Finland	
Supervisors	Prof. Elena Simona Lohan Tampere University, Finland	Prof. Antonella Molinaro University Mediterranea of Reggio Calabria, Italy
	Prof. Claudia Campolo University Mediterranea of Reggio Calabria, Italy	Dr. Aleksandr Ometov Tampere University, Finland
	Ilaria Lener T6 EcoSystems, Italy	
Pre-examiners	Prof. Enrico Natalizio University of Lorraine, France	Prof. Nele Mentens KU Leuven, Belgium
Opponents	Prof. Peeter Ellervee Tallinn University of Technology, Estonia	Dr. Valeria Loscri INRIA, Lille, France

The originality of this thesis has been checked using the Turnitin Originality Check service.

ISBN 978-952-03-3390-4 (print)
ISBN 978-952-03-3391-1 (online)
http://urn.fi/URN:ISBN:978-952-03-3391-1

Carbon dioxide emissions from printing Tampere University dissertations have been compensated.

PunaMusta Oy – Yliopistopaino
Joensuu 2024

Edge-assisted Task Offloading in the Internet of Wearable Things

Doctoral thesis submitted by **Waleed Bin Qaim** in order to be eligible for a double doctoral degree awarded by Tampere University, Finland and the University Mediterranea of Reggio Calabria, Italy.

Tampere University

Doctoral Programme in Dynamic Wearable Applications with Privacy Constraints, Tampere University

Università degli Studi di Reggio Calabria Mediterranea

Dipartimento di Ingegneria dell'Informazione, delle Infrastrutture e dell'EnergiaSostenibile (DIIES), PhD Course in Information Engineering, XXXV CICLO

AUTHOR:

Waleed Bin Qaim

Digitally signed by Waleed Bin Qaim
DN: cn=Waleed Bin Qaim, c=FI, o=Tampere University, ou=Electrical Engineering, email=waleed.binqaim@tuni.fi
Reason: I am the author of this document
Location: Tampere, Finland
Date: 2024.03.08 12:10:52 +02'00'

Waleed Bin Qaim

MAIN SUPERVISOR:

Jari Nurmi

Digitally signed by Jari Nurmi
DN: cn=Jari Nurmi, c=FI, o=Tampere University, ou=ITC Faculty, email=jari.nurmi@tuni.fi
Date: 2024.03.08 13:13:15 +02'00'

Prof. Jari Nurmi

CO-SUPERVISORS:

Simona Lohan

Digitally signed by Simona Lohan
DN: cn=Simona Lohan, c=FI, o=Tampere University, ou=ITC, email=elena-simona.lohan@tuni.fi
Date: 2024.03.11 09:54:18 +02'00'

Prof. Elena Simona Lohan

Firmato digitalmente da: Antonella Molinaro
Organizzazione: UNIVERSITAMEDITERRANEA/00163260805
Limitazioni d'uso: Explicit Text: I titolari fanno uso del certificato solo per le finalità di lavoro per le quali esso è rilasciato. The certificate holder must use the certificate only for the purposes for which it is issued.
Data: 11/03/2024 10:40:42

Prof. Antonella Molinaro

Firmato digitalmente da: Claudia Campolo
Organizzazione: UNIVERSITAMEDITERRANEA/00163260805
Limitazioni d'uso: Explicit Text: I titolari fanno uso del certificato solo per le finalità di lavoro per le quali esso è rilasciato. The certificate holder must use the certificate only for the purposes for which it is issued.
Data: 11/03/2024 12:05:21

Prof. Claudia Campolo

Aleksandr Ometov

Digitally signed by Aleksandr Ometov
Date: 2024.03.11 10:50:11 +02'00'

Dr. Aleksandr Ometov

Ilaria Lener

Firmato digitalmente da: ILARIA LENER
Data: 11/03/2024 13:19:07

Ilaria Lener

Tampere (Finland), March 2024

This dissertation is funded by the European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska-Curie grant agreement No. 813278 (A-WEAR: A network for dynamic wearable applications with privacy constraints, http://www.a-wear.eu/)

To my family

ACKNOWLEDGEMENTS

The research work presented in this thesis has been carried out at the Unit of Electrical Engineering, Tampere University, Finland and the DIIES Department, University Mediterranea of Reggio Calabria, Italy in partial fulfillment of the requirements for a double Ph.D. degree. This work was supported by the A-WEAR project funded by the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie Grant Agreement No. 813278.

I would like to express my sincere gratitude to my supervisory team including Prof. Jari Nurmi, Prof. Elena-Simona Lohan, Prof. Antonella Molinaro, Prof. Claudia Campolo, and Dr. Aleksandr Ometov for their unwavering guidance, support, and encouragement throughout my doctoral studies. I have immensely benefitted from their knowledge, expertise, and experience. The invaluable discussions, advice, and suggestions they generously provided contributed significantly towards the completion of this research work. I genuinely believe that I would have never made it this far with out their continuous support and mentorship. I also express my deep gratitude to my industrial supervisor, Ilaria Lener, for the time and energy she invested to guide me throughout, particularly during my industrial secondment at T6 EcoSystems, Italy. Thank you all for everything.

I am grateful to my pre-examiners, Prof. Enrico Natalizio (University of Lorraine, France) and Prof. Nele Mentens (KU Leuven, Belgium), for reviewing this thesis and providing highly valuable feedback and comments. I am also thankful to Prof. Peeter Ellervee (Tallinn University of Technology) and Dr. Valeria Loscri (INRIA, Lille, France), for agreeing to serve as opponents during the thesis defense.

I would like to express my heartfelt appreciation to my colleagues, Lucie Klus, Viktoriia Shubina, Asad Ali, Justyna Skibińska, Pavel Pascacio, Laura Flueratoru, Darwin Quezada Gaibor, Asma Channa, Ekaterina Svertoka, Raúl Casanova-Marqués, Sylvia Holcer, Tomas Bravenec, Salwa Saafi, Olga Chukhno, and Nadia Chukhno, with whom I had the privilege to collaborate during my doctoral studies. It has been an honor to have met and worked alongside such dedicated and amazing group of people. I wish you all the very best in life.

I would also like to convey my deep appreciation to my friends, Fahad Sohrab and Muhammad Zeeshan Waheed, for the meaningful impact they have had on my life. Their consistent support, uplifting encouragement, and cherished companionship has been a constant source of inspiration to me.

I remain highly indebted to my parents and siblings who consistently relieved me of my responsibilities and wholeheartedly encouraged me to focus on my studies. Thank you Abu, Ami, Munazza, Osama, and Ramsha (as well as my beloved nephew, Yahya), for being a source of continuous support and encouragement. I feel blessed to have you all in my life and the bond we share.

Finally, heartfelt appreciation goes to my wife, Rabia, my biggest source of motivation. She was consistently there to listen to me, believe in me, motivate me in tough days, and cheer for me in my accomplishments. It meant the world to me. Words can not do justice for her unconditional and continuous support that always kept me going. I was always amazed by her willingness to discuss my research work and proofread countless pages of my writings. It would never have been possible without you. My love and gratitude for always being by my side. Lots of love to our beautiful little daughters, Salwa and Aniyah, who have been a source of continuous happiness and relief in this challenging journey.

Waleed Bin Qaim
March 2024, Tampere, Finland.

ABSTRACT

The personal devices market is witnessing a rapid evolution due to the development of small form-factor electronics, followed by the increasing number of various value-added and entertainment applications. Personal mobile devices, also called wearables, include devices such as smartphones, smartwatches, smart glasses, smart jewelry, smart clothes, and augmented or virtual reality headsets, to name a few. These wearable devices have launched a new trend in the Internet of Things (IoT) era, namely the Internet of Wearable Things (IoWT). Wearables are small IoWT devices capable of sensing, storing, processing, and exchanging data to assist users by improving their everyday life tasks through various applications.

With the increasing consumer interest in wearables, there is a consistent demand for the development of new computationally intensive and latency-critical applications such as interactive online gaming, virtual/augmented reality, ultra-high-definition video streaming, autonomous driving, image processing, and machine learning, among others. Historically, wearable and handheld devices were not designed to execute computationally intensive operations, primarily due to constraints such as limited battery capacity and heat radiation. Consequently, the need for developing highly energy-efficient solutions has become imperative to meet the demands of the latest power-intensive wearable sensors and applications, aiming to meet user expectations. Thus, energy efficiency within wearables has emerged as a dynamic field of research.

This thesis employs a systematic literature review approach to conduct a comprehensive survey of energy-efficient solutions proposed for diverse IoWT applications. The existing research published from 2010 to 2020 is scrutinized, and a taxonomy of the available solutions is presented based on the targeted application area. Moreover, a thorough qualitative and comparative analysis of existing studies within each category is provided highlighting the merits, demerits, main performance parameters, and major contributions of each solution. Furthermore, we provided different statis-

tical analysis providing insights into the publication trends in this field of research, commonly used tools to evaluate proposed solutions, and frequently employed communication technologies in wearables. Additionally, a detailed discussion is provided outlining the predominant approaches found in the literature for enhancing energy efficiency in wearables while also emphasizing the challenges involved.

While wearables have the potential to completely revolutionize everyday life of individuals, they have also brought a plethora of new challenges for the research and industrial community to address. These challenges include increasing demand for enhanced computational power, improved communication capabilities, enhanced security and privacy features, reduced form factor, minimal weight, and better user comfort. Many of these challenges stem from the limited battery power and insufficient computation resources available on wearable devices.

In such a context, task offloading is a technique that allows wearables to leverage the resources available on nearby edge devices to not only conserve the wearable's limited resources but also improve its computational capacity to enhance the overall user experience. Therefore, this thesis presents a numerical analysis of task offloading for wearables in a two-tier edge architecture, considering different task offloading scenarios from the wearable to a device located at the network edge. Such a device could be a smartphone paired with the wearable or an edge server co-located with the cellular base station. A comprehensive performance evaluation conducted under a wide variety of realistic settings in terms of task demands, processing capabilities, and data rate, is provided unveiling the circumstances in which offloading is convenient and when it is not, in terms of meaningful metrics.

Subsequently, this thesis proposes a framework for Reinforcement Learning (RL)-based task offloading in the IoWT. The task offloading process is formulated considering the tradeoff between energy consumption and task accomplishment time. Moreover, we model the task offloading problem as a Markov Decision Process (MDP) and utilize a model-free Q-learning technique of RL to enable the wearable device to make optimal task offloading decisions without prior knowledge. Furthermore, we evaluate the performance of the proposed framework through extensive simulations for various applications and system configurations conducted in the ns-3 network simulator. We also show how varying the main system parameters of the Q-learning algorithm affects the overall performance. Finally, as part of the conclusion, we also highlight some potential future research directions.

SOMMARIO

Il mercato dei dispositivi personali sta assistendo a una rapida evoluzione dovuta allo sviluppo di componenti elettronici di piccole dimensioni, seguita dal numero crescente di varie applicazioni a valore aggiunto e di intrattenimento. I dispositivi mobili personali, chiamati anche dispositivi indossabili, includono dispositivi come smartphone, smartwatch, occhiali intelligenti, gioielli intelligenti, vestiti intelligenti e visori per realtà aumentata o virtuale, solo per citarne alcuni. Questi dispositivi indossabili hanno lanciato una nuova tendenza nell'era IoT, ovvero IoWT. I dispositivi indossabili sono piccoli dispositivi in grado di rilevare, archiviare, elaborare e scambiare dati per assistere gli utenti migliorando le loro attività quotidiane attraverso varie applicazioni.

Con il crescente interesse dei consumatori per i dispositivi indossabili, esiste una domanda costante per lo sviluppo di nuove applicazioni ad alta intensità di calcolo e critiche in termini di latenza, come giochi online interattivi, realtà virtuale/aumentata, streaming video ad altissima definizione, guida autonoma, elaborazione di immagini, e apprendimento automatico, tra gli altri. Storicamente, i dispositivi indossabili e portatili non sono stati progettati per eseguire operazioni ad alta intensità di calcolo, principalmente a causa di vincoli quali la capacità limitata della batteria e il surriscaldamento. Di conseguenza, la necessità di sviluppare soluzioni ad alta efficienza energetica è diventata indispensabile per soddisfare le richieste dei più recenti sensori e applicazioni indossabili energivori, con l'obiettivo di soddisfare le aspettative degli utenti. Pertanto, l'efficienza energetica nei dispositivi indossabili è emersa come un ambito di ricerca in evoluzione.

Questa tesi utilizza un approccio di revisione sistematica della letteratura per condurre un'indagine completa di soluzioni efficienti dal punto di vista energetico proposte per diverse applicazioni IoWT. Viene esaminata attentamente la ricerca esistente pubblicata dal 2010 al 2020 e viene presentata una tassonomia delle soluzioni disponibili in base all'area di applicazione di interesse. Inoltre, viene fornita un'analisi

qualitativa e comparativa approfondita degli studi esistenti all'interno di ciascuna categoria evidenziando i meriti, i demeriti, i principali parametri di prestazione e i principali contributi di ciascuna soluzione. Inoltre, sono state fornite diverse analisi statistiche che forniscono approfondimenti sulle tendenze di pubblicazione in questo campo di ricerca, sugli strumenti comunemente utilizzati per valutare le soluzioni proposte e sulle tecnologie di comunicazione frequentemente utilizzate nei dispositivi indossabili. Inoltre, viene fornita una discussione dettagliata che delinea i principali approcci presenti in letteratura per migliorare l'efficienza energetica nei dispositivi indossabili, sottolineando al tempo stesso le sfide associate.

Se da un lato i dispositivi indossabili hanno il potenziale per rivoluzionare completamente la vita quotidiana degli individui, dall'altro comportano anche una serie di nuove sfide da affrontare per la ricerca e la comunità industriale. Queste sfide includono la crescente domanda di maggiore potenza di calcolo, migliori capacità di comunicazione, funzionalità avanzate di sicurezza e privacy, dimensioni ridotte, peso minimo e migliore comfort per l'utente. Molte di queste sfide derivano dalla potenza limitata della batteria e dalle risorse di calcolo insufficienti disponibili sui dispositivi indossabili.

In un tale contesto, l'*offloading* dei *task* computazionali è una tecnica che consente ai dispositivi indossabili di sfruttare le risorse disponibili sui dispositivi vicini alla periferia della rete, non solo per conservare le risorse limitate del dispositivo indossabile, ma anche per aumentarne la capacità computazionale al fine di migliorare l'esperienza complessiva dell'utente. Pertanto, questa tesi presenta un'analisi numerica dell'offloading dei task per i dispositivi indossabili in un'architettura edge a due livelli, considerando diversi scenari in cui i task vengono delegati dal dispositivo indossabile a un dispositivo situato all'edge della rete. Un dispositivo di questo tipo potrebbe essere uno smartphone abbinato al dispositivo indossabile o un server periferico co-locato con la stazione base cellulare. Viene fornita una valutazione completa delle prestazioni, in termini di metriche significative e condotta in un'ampia varietà di contesti realistici in termini di richieste di task, capacità di elaborazione e velocità di trasmissione dei dati, individuando le circostanze in cui l'offloading è conveniente e quando non lo è.

Successivamente, questa tesi propone un framework per l'offloading di task basato su RL in IoWT. Il processo di offloading dei task è formulato considerando il compromesso tra il consumo energetico e il tempo di esecuzione del task. Inoltre, si modella il problema del task offloading come un MDP e utilizziamo una tecnica di Q-learning priva di modelli di RL per consentire al dispositivo indossabile di prendere decisioni ottimali senza conoscenze preliminari. Inoltre, si valutano le prestazioni del framework proposto attraverso simulazioni approfondite per varie applicazioni e configurazioni di sistema condotte nel simulatore di rete ns-3. Si mostra anche come la variazione dei principali parametri di sistema dell'algoritmo Q-learning influisce sulle prestazioni complessive. Infine, per concludere vengono evidenziate anche alcune potenziali direzioni di ricerca future.

CONTENTS

List of Tables

ABBREVIATIONS

3D Three Dimensional

AI Artificial Intelligence

AR Augmented Reality

BAN Body Area Network

BLE Bluetooth Low Energy

CNN Convolutional Neural Network

DRL Deep Reinforcement Learning

EH Energy Harvesting

ECG Electrocardiography

EEG Electroencephalography

EMG Electromyography

HVAC Heating, Ventilation, and Air Conditioning

IEEE Institute of Electrical and Electronics Engineers

IoT Internet of Things

IoV Internet of Vehicles

IoWT Internet of Wearable Things

KM Kuhn-Munkras algorithm

KPI Key Performance Indicator

LTE Long-Term Evolution

M2M Machine-to-Machine

MAC Medium Access Control

MANET Mobile Ad Hoc Network

MCC Mobile Cloud Computing

MCSS Multicombined Computing Sorting Segmentation

MDP Markov Decision Process

MEC Mobile Edge Computing

MEMS Miniature Micro-Electro-Mechanical

MR Mixed Reality

ML Machine Learning

NB-IoT Narrowband IoT

QoE Quality-of-Experience

QoS Quality-of-Service

RFID Radio-Frequency Identification

RL Reinforcement Learning

RMSE Root Mean Square Error

SLR Systematic Literature Review

SNR Signal to Noise Ratio

SoC System on Chip

STLF State Loss Function

SYLF System Loss Function

TAP Task Assignment Problem

TDMA Time-Division Multiple Access

UAV Unmanned Aerial Vehicle

V2V Vehicle-to-Vehicle

VR Virtual Reality

WBAN Wireless Body Area Network

WBSN Wearable Body Sensor Network

Wi-Fi Wireless Fidelity

WIoT Wearable Internet of Things

XR Extended Reality

1 INTRODUCTION

This chapter serves as an introduction to the topic, outlining the motivation behind this research. It presents the main research questions we aimed to address through this thesis followed by the contributions it makes. Finally, it provides an overview of the thesis structure.

1.1 Motivation

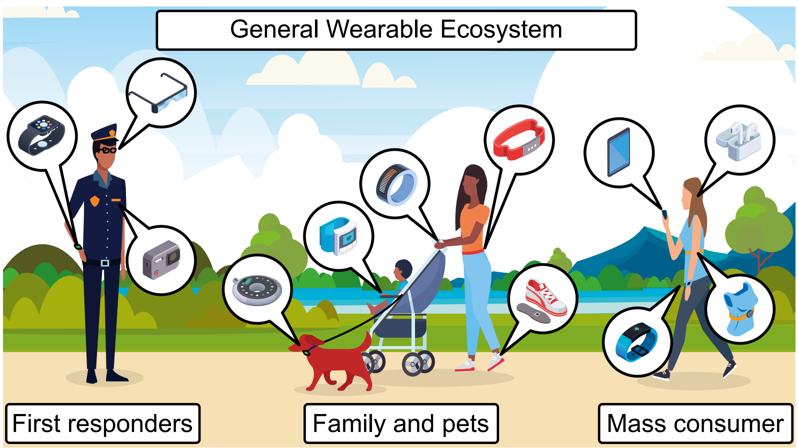

Figure 1.1 Forthcoming wearable integration scenario [1].

The emergence of compact, affordable, and battery-powered computing components, such as microprocessors and microcontrollers, has paved the way for designing a diverse range of small-scale devices that can connect both with each other and to the Internet. These small form-factor devices serve as the foundation for the IoT concept [2]. This technological evolution has enabled millions of objects to establish seamless real-time communication with the Internet, enhancing accessibility and control. More recently, a new paradigm known as the IoWT has emerged, focusing on smart devices that are worn or carried near the body [3].

The IoWT, also known as the Wearable Internet of Things (WIoT) [4], involves a fusion of diverse smart wearable devices, as illustrated in figure 1.1. These include smartwatches, wristbands, smart shoes, smart jewelry, smart glasses, adhesive skin patches, and more. These wearables come equipped with an array of sensors, computational units, and communication modules, enabling them to continuously sense, process, and exchange various types of data [5].

Another closely related research domain is Wearable Body Sensor Networks (WBSNs) [6] also commonly known as, Wireless Body Area Networks (WBANs) [7]. These fields primarily focus on applications related to human health, which shares some common ground with the IoWT. However, there exists a slight dissimilarity between WBSNs/WBANs and IoWT in terms of the number of sensors or devices involved. WBSNs or WBANs typically aim to incorporate a larger number of wearable sensor nodes, often up to 50 nodes, forming a network that collaboratively works toward a common goal. For instance, multiple wearable sensor nodes collaboratively monitoring an individual's overall health. Whereas, IoWT devices are typically standalone units that are presently being used for a wide range of applications, including health monitoring, human activity recognition, tracking and localization, as well as various gaming and entertainment gadgets [8], [9]. Furthermore, wearables enhance user convenience and efficiency in everyday tasks by providing visual and auditory alerts, such as incoming calls and messages, delivering weather updates, and displaying essential real-time information [10] to name a few. These capabilities have the potential to revolutionize everyday human activities, contributing to an overall improvement in quality of life [11], [12].

Currently, there is a significant surge in consumer fascination with wearable devices with exponential growth anticipated for upcoming years. A recent analysis of market trends indicates that the wearable technology sector is forecasted to reach more than 150 billion EUR by 2028 [13]–[15].

Nonetheless, the transition in emphasis from traditional smartphones to intelligent wearables has introduced an array of research challenges that need to be tackled by the scientific, research, and industrial sectors. In addition to the expanding and diverse application domains, there is also a surge in the need for enhanced wearable performance. Presently, wearables encounter various limitations including limited battery lifetime, computational and communication capability, security and privacy features, physical design, weight, and user comfort, among other factors [16], [17].

The demand for highly energy-efficient solutions has surged in parallel with the advancements in wearable technology and the growing user interest in wearables for a diverse range of value-added and entertainment applications. Furthermore, the frequent need to recharge personal devices and electronic gadgets has become increasingly burdensome and inconvenient for users. As a result, energy efficiency in wearables has emerged as the prime focus in research. Although, there has been a significant advancement in designing efficient batteries to extend device battery life; there is also a concurrent increase in the demand for enhanced processing power and the complexity of applications. Hence, the primary challenge originates from the constrained computational capabilities and limited battery life of these devices [16], [18], which limit their utility. Therefore, the development of highly energy-efficient solutions has become crucial to meet the requirements of the latest power-intensive wearable sensors and applications, catering to the evolving user demands.

Over the years, numerous techniques have been proposed in the literature geared towards enhancing the computational capabilities as well as improving energy efficiency of wearable devices [1]. Mobile Edge Computing (MEC) has recently emerged as a promising solution that enables mobile devices with limited resources to use task offloading to leverage the high energy, computation, and storage capabilities of nearby devices such as more powerful smartphones as well as standalone edge servers or those co-located with Wi-Fi Access Points (APs) / Cellular network Base Stations (BSs) [19], [20]. Task offloading is the process of transferring input data for a task, initially created on the wearable device, to a nearby computing entity with enhanced resources for processing. Subsequently, the processed result is returned to the wearable device where the task originated [21]. Task offloading not only improves the energy efficiency of resource constrained devices such as a wearables but also brings additional advantages such as enhanced storage and computational capabilities beyond energy conservation. However, the benefits achieved are not usually obvious and necessitate a case-by-case analysis to determine *when, what, and where* to offload. Different applications may have varying computation/communication requirements. For instance, there can be scenarios where the energy spent by the wearable in transferring data to the task executor node might exceed the energy used for local computation, rendering task offloading inefficient. Similarly, certain conditions can lead to an increase in the overall task accomplishment time when offloading tasks. For instance, it can occur when transferring large input data across dynamic/low-capacity wireless connections [22]. Therefore, it is essential to accurately estimate the benefits of task offloading in terms of energy consumption and its alignment with the latency requirements of diverse applications.

Additionally, the integration of Artificial Intelligence (AI) into Edge Computing has recently emerged as an active research area [23]. Particularly, Machine Learning (ML) approaches like RL can provide promising self-learning solutions. Unlike the conventional supervised and unsupervised ML techniques, in RL the agent does not need to be trained with extensive training data samples to predict the output of new inputs. Rather, leveraging Q-learning, a RL-based technique, can enable embedding intelligence into IoWT devices, such as wearables, to allow them to iteratively learn from their own experiences through trial and error, to make optimal task offloading decisions in varying situations based on a predefined reward function without prior knowledge [24].

1.2 Research Questions

Overall, the goal of this thesis is to advance the state of knowledge in energy efficiency optimization for the IoWT technology by investigating the current research landscape. Moreover, it includes identifying the fundamental challenges inherent to the wearable technology development. Furthermore, it involves exploring the potential benefits and limitations of efficient computation techniques such as task offloading, and leveraging Q-learning, to enhance the overall performance.

In this context, we set forth the following research questions to be addressed in this thesis:

Q1. What is the current state of research focused on energy efficiency in the IoWT technology, including year-wise publication trends, main application areas, performance parameters, evaluation tools, prevalent wireless communication technologies, and strategies for enhancing energy efficiency?

Q2. What are the potential benefits and limitations of task offloading for wearables in multi-tier edge architectures in terms of task accomplishment time and energy consumption, considering realistic settings regarding computing task requirements, device capabilities, and inter-device distance?

Q3. How can Q-learning, a RL-based technique, be effectively utilized to optimize task offloading for wearables in an edge computing framework to enhance their overall performance, network resource utilization, and end-user experience?

1.3 Contributions

The detailed contributions this thesis provides are organized in 3 chapters, each corresponding to a research question from the list presented in section 1.2. Chapter 1 (C1), focuses on the energy efficiency aspect of wearable devices in the IoWT. Chapter 2 (C2), quantifies the benefits of task offloading for wearables in an edge-assisted architecture. Chapter 3 (C3), explores the potential application of ML to optimize the task offloading process on wearable devices.

C1. A Systematic Literature Review (SLR) on Energy Efficiency in the IoWT
The main contributions (detailed in chapter 2) are summarized as follows:

- A taxonomy of the IoWT solutions focused on energy efficiency is presented, classifying them into four categories: healthcare, activity recognition, smart environments, and general solutions, based on the targeted application area.
- A qualitative and comparative analysis of prior research solutions is provided, highlighting their merits, demerits, key performance metrics, and major contributions.
- A statistical analysis of the available solutions is provided interms of year-by-year publications, application areas, evaluation mechanisms, simulation platforms, and communication technologies.
- A summarizing discussion is presented regarding the main approaches adopted in the literature for enhancing energy efficiency in wearables, highlighting the benefits and associated challenges.

C2. Task Offloading for Wearables in a Two-Tier Edge Architecture
The main contributions (detailed in chapter 3) are summarized as follows:

- A two-tier edge architecture for task offloading from wearables to the edge is presented that includes both a smartphone and an edge server as potential task executors.
- An in-depth analysis of task execution performance on wearable devices is provided, identifying the boundaries and constraints that come into play.

- A detailed discussion is provided on the conditions under which task offloading to the edge can improve performance and to what extent. We investigate two important metrics, i.e., task accomplishment time and energy consumption due to the computational and communication processes of mobile devices involved in the task offloading procedure. A mathematical formulation is proposed for these metrics to present an analysis that provides a comprehensive and adaptable analytical playground, including a wide array of practical scenarios that factor in computing task requirements, device capabilities, and the spatial separation between the entities involved.

C3. Reinforcement Learning-based Task Offloading for Wearables
The main contributions (detailed in chapter 4) are summarized as follows:

- An edge computing framework is proposed involving a wearable device paired to the user's smartphone (acting as an edge node for the wearable) to enhance the overall user experience by optimizing task accomplishment time and energy consumption of the battery-powered devices involved.

- A mathematical formulation is provided for deriving the desired performance metrics, i.e., task accomplishment time and energy consumption for local computation and offloading scenarios.

- The task offloading procedure is formulated as a MDP and a model-free Q-learning-based algorithm for task offloading is proposed that adapts to the variations in network dynamics to make the best possible use of computation resources in the system.

- The performance analysis of the proposed algorithm is provided in terms of different parameters including average task accomplishment time, average energy consumption, percentage of tasks offloaded, and total cost. The analysis is based on extensive simulations performed in the ns-3 Network Simulator that utilizes realistic communication models. The simulations are carried out for multiple applications under a wide variety of realistic settings while also showing how varying main system parameters of the Q-learning algorithm affects overall performance.

1.4 Thesis Outline

This thesis is organized into 5 chapters. A brief description regarding the content of each chapter is as follows:

- **Chapter 1** provides an introduction to the topic, the motivation behind this research, main research questions to be addressed, and the thesis outline.
- **Chapter 2** addresses research question (Q1) by presenting a SLR of state-of-the-art solutions aiming to improve the energy efficiency of wearable devices in the IoWT.
- **Chapter 3** addresses research question (Q2) by presenting a numerical analysis of the benefits task offloading can bring to wearables in terms of task accomplishment time and energy consumption in a two-tier edge architecture.
- **Chapter 4** addresses research question (Q3) by providing details of the proposed RL-based framework enabling wearables to make intelligent task offloading decisions.
- **Chapter 5** draws the major conclusions while also highlighting avenues for future research.

  Bibliography is included at the end of this thesis.

2 SYSTEMATIC LITERATURE REVIEW

This chapter focuses on the SLR of state-of-the-art solutions aiming to improve the energy efficiency of wearable devices in the IoWT.

In section 2.1, we provide a brief background on the concept of energy efficiency for wearables and the significance of this work. We then provide details on the contributions made through this work followed by the research methodology adopted.

Section 2.2 provides a classification of the existing works as well as presents a performance, statistical, and qualitative analysis. Moreover, for each classification category, we present a consolidated summary to give the merits and demerits of each work as well as define the main performance parameters considered.

Section 2.3 presents the main strategies available in the literature to enhance energy efficiency of wearable devices while also highlighting their benefits and limitations.

Finally, section 2.4 concludes this chapter by providing a summary of the findings based on this SLR.

2.1 Background

Wearables are still facing numerous challenges. However, the main constraint continues to be the limited battery lifespan of these devices, as highlighted in various research studies [18], [25], [26]. Consequently, the development of energy-efficient solutions for these devices becomes critically important to extend the battery lifetime of wearable devices while simultaneously achieving the desired performance of their applications.

Following the increasing interest of the consumer market towards wearable technology, there have been substantial contributions from the scientific and research community. Continuous attempts are being made to design highly efficient solutions aiming to tackle the related challenges and exploit the full potential of wearable technology. Consequently, several surveys have been carried out in the field providing insights on wearable computing evolution as one of the potential solutions to solve the energy efficiency challenge inherent to wearables.

For example, Seneviratne et al. [8] presented a survey and categorization of various commercially available wearable devices based on their functionality and ease of wear. It presents limited strategies, such as battery advancements, efficient sensing, and Energy Harvesting (EH). Similarly, Tifenn et al. [27] provided a survey of energy-efficient techniques for wearable sensor networks. However, the emphasis is confined to health-related human context recognition applications. Additionally, Williamson et al. [25] presented the energy challenges for wearable sensing focusing on the Miniature Micro-Electro-Mechanical (MEMS)-based inertial measurement units. Further, Sun et al. [28] provided a survey of the enabling communication technologies that can facilitate wearable devices for contemporary and prospective applications.

Moreover, some surveys concentrate on the use of wearables for a specific application such as health monitoring [29], activity recognition [30], [31], assisted living [32], mobile crowdsensing [33], smart garments [34], and indoor positioning [35].

Among the aforementioned studies, none of them are explicitly focusing on the energy efficiency aspect, rather briefly mention the challenge. Furthermore, a statistical analysis of recent advancements in the domain of energy efficiency within the context of IoWT was missing. Therefore, we provide a comprehensive review of the state-of-the-art energy efficiency solutions for wearables employing the SLR methodology to bridge this gap in IoWT technology.

2.1.1 Contributions

The main contributions we provide in this chapter are briefly reiterated as follows:

- We presents a taxonomy of the IoWT solutions based on the targeted application area.

- We presents a detailed qualitative and comparative analysis of existing solutions.
- We presents an insightful statistical analysis of the available solutions.
- We present a summarizing discussion regarding the main energy efficiency techniques available in the literature.

2.1.2 Research Methodology

In this chapter, we follow the PRISMA guidelines, proposed in [36] as our research methodology to carry out this SLR.

The first step was to determine the appropriate keywords and associated synonyms to create a search expression. After a brief research of the literature, the following search expression was formed:

> ("energy efficien*" OR "energy conserv*" OR "low power")
> AND (wearable*)
> AND (edge OR cloud OR fog OR approximate OR IoT OR "Internet of Things" OR performance)

A search was conducted with the identified keywords for the 2010 – 2020 period in the two most widely accepted research databases in Information and Communication Technology (ICT) domain, namely Scopus [37] and Web of Science [38]. We collected a set of 2370 potentially relevant publications (as of July 2020), excluding grey literature, pre-prints, and duplicates. We then scrutinized the titles, keywords, and abstracts of the publications to identify articles that described at least topics related to energy efficiency/consumption in the IoWT field. The following exclusion criteria were formulated to refine the search results during the paper titles and abstracts' initial screening:

C1. Not related to wearable networks/computing;

C2. Pure survey and review articles;

C3. Works with no technical content;

C4. Full text not available.

The complete selection process is given in figure 2.1. After applying the refinement procedures, we reduced the articles number to 50 potentially relevant papers. After studying the selected literature references and citations, we increased the number to 151 works to be included in this systematic review focused on energy efficiency in the IoWT.

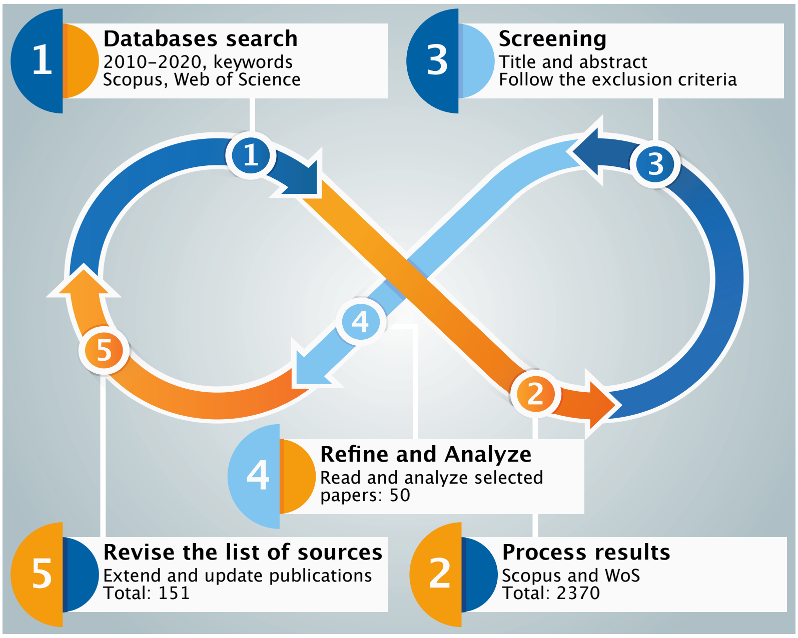

Figure 2.1 Main steps involved in the executed systematic literature review process [1].

2.2 Classification of Existing Applications and Related Technologies

An analysis of the selected papers is presented in this section, including a classification, statistical analysis, and qualitative analysis. The year-wise distribution is provided in figure 2.2. An increasing trend in the number of publications can be observed in the IoWT domain while some studies from 2019 and 2020 may still be not indexed or under review.

The selected papers were categorized into four main divisions, namely, healthcare, activity recognition, smart environments, and general solutions, based on the targeted application area. A variety of applications benefit from IoWT energy-efficient technologies in each category, as depicted in figure 2.3.

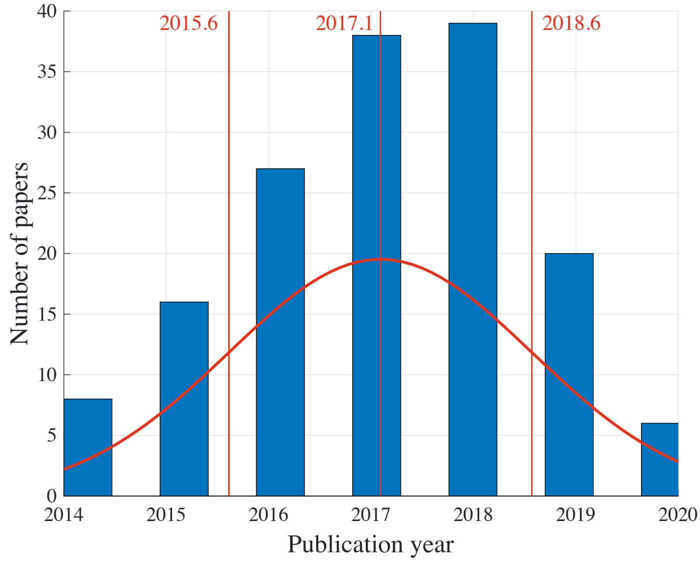

Figure 2.2 Year-wise distribution of number of articles analyzed: Red Lines correspond to mean and mean +/- standard deviation [1].

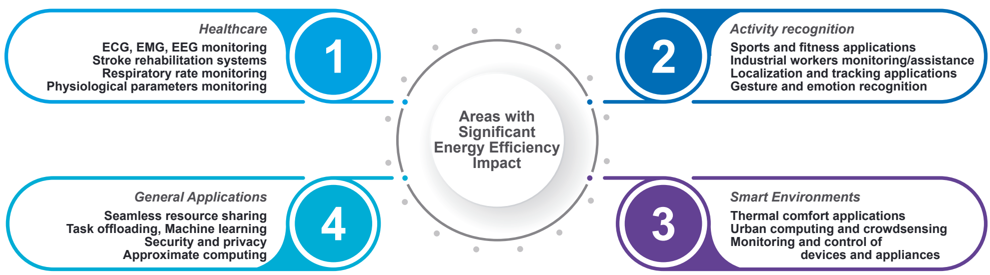

Figure 2.3 Main application domains of wearables with high energy efficiency impact [1].

Figure 2.4 shows a statistical analysis of various application areas in wearable technology. A significant portion of research articles falls within the realm of the *Healthcare* category, constituting approximately 48% of the total. Because numerous wearable devices have historically emerged in the healthcare sector [39] designed for continuous patient monitoring. In this context, wearables have been predominantly employed for tracking essential human physiological indicators. This prevailing pattern can be attributed to the initial development of wearables for specific medical applications, including continuous Electrocardiography (ECG) monitoring of heart activity, auditory aids for individuals with hearing impairments, and robotic limbs

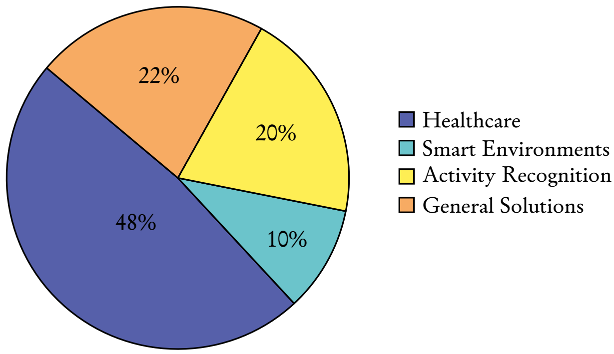

Figure 2.4 Percentage of works targeting each application area in the analysed literature [1].

for the medically paralyzed patients, etc.

Clearly, wearables have progressively extended their utility beyond the healthcare sector. For instance, research centered on human activity recognition has made a substantial contribution, accounting for up to 20% of the total. Activity recognition is primarily used to observe and record physical movements of individuals. For example, wearables have seen an increasing adoption in delivering diverse services based on user activities, such as suggesting points of interest, offering fitness guidance via step counting, and monitoring user's sports activities [40], [41]. Similarly, wearables have also facilitated location-based services [42], applications involving gesture recognition [43], and the monitoring of industrial laborers [44]. Therefore, all research works utilizing wearables to track user activities are categorized under the umbrella of *Activity Recognition*.

Numerous research studies have presented general IoT-based solutions, employing wearables that can be utilized across various domains. These solutions account for approximately 22% of the total. Likewise, wearables have also been integrated into other application domains, including Smart Environments, representing a share of around 10%.

Furthermore, wearables have discovered practical applications within the domain of *Smart Environments*. For instance, the incorporation of wearables has been proposed in the context of smart buildings, aiming to oversee and curtail electricity consumption by autonomously deactivating unnecessary electrical devices through real-time monitoring of occupants and environmental conditions [45], [46]. Similarly, certain studies employ wearables to evaluate and optimize users' thermal comfort levels within smart environments by continuous temperature monitoring and automated management of heating/cooling systems [47]. Additionally, wearables have the potential to empower individuals with disabilities, enhancing their ability to independently carry out daily activities [48]. Hence, all research works utilizing wearables for applications within the concept of smart environment are organized under this category.

Lastly, a subset of studies either do not explicitly specify the targeted application domain or are adaptable to multiple areas of application. Such studies are categorized under the label of *General Solutions*. In these cases, the researchers generally elaborate on the technology, enablers, or methodologies, without emphasis on any singular application domain.

The following subsections present a discussion on characteristics along with qualitative and performance analysis of the papers falling under each category. We provide an overview of the identified solutions by presenting the aim, merits, demerits, and major findings for each application area in tables 2.1, 2.3, 2.5, and 2.7. A comparative assessment of solutions is provided in terms of the main performance parameters for each application area including traditional common Key Performance Indicators (KPIs) parameters, e.g., latency, energy consumption, and throughput, as well as specific ones in Tables 2.2, 2.4, 2.6, and 2.8.

It is important to highlight that certain application areas use performance parameters tailored to their specific contexts. For instance, in the healthcare domain, emphasis is on *Signal Reconstruction Quality*, which gauges the precision of signal reconstruction at the gateway node based on sensor-recorded observations. Another pertinent parameter is *Network Lifetime*, which denotes the operational duration of a network until one of its nodes depletes its energy. Moreover, *Accuracy* measures the capability of a wearable device to predict the occurance of a specific disease. Furthermore, *Signal to Noise Ratio (SNR)* quantifies the relative strength of the desired signal in relation to the noise level. Additionally, *Compression Ratio* serves as a metric showcasing the extent to which a dataset is compressed. Lastly, *Reliability* is given as the probability of encountering failure within the system.

Furthermore, in activity recognition applications, the *Sensitivity* metric indicates a wearable device's capability to promptly detect user activities. In the context of smart environments, the parameter of *Video Quality* is employed for crowdsensing applications. Lastly, within the domain of general solutions, both *Execution Time* and *Transmission Time* are considered, serving as indicators of the speed at which a wearable device conducts computations and transmits data to a gateway node.

The following subsections present a comprehensive overview of the applications along with the related performance metrics.

2.2.1 Healthcare Applications

Table 2.1 Summary of recent studies in healthcare domain [1].

Ref.	Aim of study & Major findings	Merits	Demerits
[49] 2020	Real time compressive sensing-based recovery of the ECG signals at the IoT gateway using multicore processors	Improved latency, privacy and energy efficiency; independent on cloud infrastructures	Only suitable for sparse signals
[50] 2019	An IoT architecture relaying on open standards (oneM2M and openEHR) and allowing for the interoperability between different devices and software to track physiological parameters of patients in emergency wards	Interoperability, low latency, low cost, enhanced battery lifetime, efficient ESP8266 Wi-Fi nodes	No real-time validation, high latency with deep sleep states
[51] 2019	An energy-efficient data-criticality aware routing protocol for WBANs	Enhanced network lifetime, emergency data delivery	No mobility support, single point of failure
[52] 2019	A wearable cardiovascular healthcare system with cross-layer optimization comprising an efficient sensing patch with embedded signal denoising, data compression, and data transmission capabilities	Miniaturized footprint, low power consumption, embedded signal processing capability	Low accuracy on the mobile device side
[53] 2018	A wearable ring sensor for monitoring autonomic nervous system activities	Small size, ease of use, low cost, mobile application	No comparison with other devices
[54] 2018	A MDP-based transmission strategy for multi-hop intra-BAN communication	Adaptive transmission power optimization	Limited performance comparisons
[55] 2018	An efficient next-hop node selection framework based on multi-parameter path cost function WBAN	Energy-efficient, low packet loss, high throughput and extended network lifetime	Control messages overhead, human body movement not considered

Ref.	Aim of study & Major findings	Merits	Demerits
[56] 2018	A wrist-worn ECG sensor measuring heart rate variability in out-of-the-clinic settings with Three Dimensional (3D) printed elements for personalization, capable of integrating with the Azure IoT system	Low power, low weight, personalized features	Low accuracy
[57] 2018	A wearable ring sensor with an iOS application for remotely monitoring parameters of patients, e.g., electrodermal activity, heart rate, locomotion, temperature	Compact and miniaturized design, low cost, recording various biosignals from user's finger, high accuracy	High motion artifacts
[58] 2018	An IoT-based smart wearable armband for stroke rehabilitation system deploying ML algorithms to strengthen the motion patterns	Mobility support, small size, real-time feedback of muscle activities, personalization with 3D printed robotic hand	Tested on a single subject, limited gesture recognition supported
[59] 2018	A mobile real-time health monitoring architecture based on a heterogeneous multicore platform for ECG signal processing	Enhanced battery life, low latency, low power device design	Sub-optimal performance for clinical-grade signals due to frequent transmissions
[60] 2017	A dictionary-based lossy signal compression technique for enhancing energy efficiency of wearables	Energy-efficient, high compression efficiency	High computation cost
[61] 2017	A low-power wearable device for continuous respiratory rate monitoring using a three-axis accelerometer from the sternum with an integrated motion artifact rejection algorithm	Efficient motion artifact rejection to remove noisy data	Limited mobility, not easy-to-use, limited battery life, no real-world testing
[62] 2017	A data-driven compressive sensing framework that can learn signal characteristics and personalized features from physiological signals	Low computational complexity, improved compression ratio	No real-time validation provided
[63] 2017	A compressed sensing-based multichannel EEG monitoring system with efficient signal compression and recovery	No prior knowledge of the signal sparsity required, improved reconstruction quality, robust	No real-time validation provided
[64] 2016	A 6LoWPAN-enabled WBAN platform with 6 different biomedical sensors optimized to meet the QoS requirements for healthcare applications	High throughput, low power, scalability, interoperability, low latency	Vulnerability to obstacles, high packet collisions
[65] 2015	An IEEE 802.15.4 based QoS design for WBAN MAC layer with beacon mode deploying tree topology supporting high-priority data transmissions	Energy-efficient, incorporating data priority feature for critical data	Starvation problem faced by low priority nodes not considered

Ref.	Aim of study & Major findings	Merits	Demerits
[66] 2016	A multihop WBAN configuration approach by creating a virtual cluster to allocate slots for simultaneous transmissions by using a multi-channel TDMA approach for wearable M2M systems	Enhanced throughput, low power consumption, low latency, better scalability	Initial setup time increases exponentially with number of nodes
[67] 2017	A configurable bio-signal acquisition wearable device for real-time monitoring on an IoT based web interface with a balanced trade-off between energy efficiency and data transmission rate	High data rate, low energy consumption, compact design	Bulky, not easy-to-use
[68] 2015	A web-based motion detection system for healthcare	Real-time bidirectional communication	High resource consumption, false alarms, lack of analysis
[69] 2018	A patient monitoring systems deploying relay-based task offloading decision model with the efficient recipient selection function	Low path loss, high computation capacity, locally processed packets	No experimental validation
[70] 2016	A wearable armband with a mobile application for unobstructed measurement of the ECG signal	Multiple activities support (sitting, hand movement, jogging, and running)	No validation on multiple subjects, high error rate
[71] 2016	A 3D Ray Launching deterministic simulation tool for feasibility and performance optimization of the WBAN-based e-Health systems within complex indoor scenarios	Low processing time, high accuracy, optimal estimation of number and position of transmitters	Patient mobility not considered
[72] 2015	A Cyber-Physical System for remote monitoring of old age home residents in real-world scenarios	Secure, scalable, low power, low cost, easy deployment	High latency, increased energy consumption due to the MAC retry attempts

In recent times, the emergence of IoWT technology along with advancements in wireless communication has significantly transformed the medical field [73]. Additionally, the process of sensor miniaturization has facilitated the creation of numerous intelligent healthcare devices that seamlessly integrate ease-of-use and portability while adding the ability to connect to the Internet to access cloud services. These include wearable devices designed for continuous patient monitoring within hospital settings [50], as well as compact gadgets engineered to continuously detect and monitor diverse health indicators of individuals throughout their daily routines [56]. The healthcare application domain of wearables covers an array of solutions, including systems for monitoring heart and respiratory rates [61], stroke rehabilitation systems [58], and the observation of heart, muscle, and brain activities through signals such as ECG, Electromyography (EMG), and Electroencephalography (EEG) [74]. Various wearable devices have been developed to consistently sense and measure a spectrum of physiological parameters in both humans and animals, including heart rate, blood pressure, body temperature, and stress hormones among others [57].

The concept of IoWT facilitates the realization of remote patient monitoring systems, wherein patients utilize one or more wearable devices for ongoing health surveillance. These devices consistently track the patient's well-being and store the collected data within online databases, allowing the patient's healthcare provider to evaluate the information. Additionally, automated assistance mechanisms are being explored for emergency scenarios. For instance, in a critical situation, a call could be automatically initiated to the caregiver or medical staff for prompt intervention [75].

Analysis of table 2.2 reveals that performance monitoring by most of studies focus on energy consumption, followed in order by accuracy, latency, and throughput. Notably, reliability appears to be the least explored within the domain of wearable healthcare applications. Additionally, given the continuous monitoring of diverse physiological parameters in most healthcare wearables, there is a tendency for these devices to deplete their energy resources due to the extensive sensing process, leading to redundant data generation and prolonged data processing times. Consequently, approaches such as compressive sensing and data compression emerge as highly effective strategies for conserving energy in healthcare applications [76]. A more comprehensive discussion of these strategies is presented in section 2.3.

2.2.2 Activity Recognition Applications

In recent years, the utilization of wearables has progressively extended into activity recognition applications [77]–[79]. This expansion has been facilitated by the miniaturization of electronic components, which has enabled the integration of multiple sensors within a single wearable device, including accelerometers, gyroscopes, magnetometers, heart-rate sensors, and more. Such sensors are capable of detecting diverse human activities [80]. Numerous applications heavily rely on the continuous monitoring and documentation of human activities, spanning domains such as assessing fitness levels through sports activities, identifying frequently visited locations, detecting falls, monitoring sleep patterns and fatigue, recognizing gestures and emotions, managing household tasks, and beyond [81]–[83]. Similarly, wearables have also

Table 2.2 Main parameters considered by recent studies in healthcare domain [1].

Ref.	Energy Consumption	Signal reconstruction quality	Latency	Network Lifetime	Throughput	Accuracy	SNR	Compression ratio	Reliability
[49]	✔	✔							
[50]	✔		✔	✔					
[51]			✔	✔	✔				
[52]	✔					✔	✔	✔	
[53]	✔					✔			
[54]	✔				✔				
[55]	✔			✔	✔				
[56]						✔			
[57]						✔			
[58]						✔			
[59]	✔							✔	
[60]	✔					✔		✔	
[61]						✔			
[62]		✔					✔		
[63]		✔							
[64]	✔		✔		✔				
[65]			✔		✔				
[66]	✔		✔		✔				
[67]	✔				✔				
[68]	✔		✔						
[69]	✔		✔	✔	✔				
[70]						✔			
[71]							✔		
[72]									✔

found utility in tracking occupational activities and enhancing worker performance within various work environments [84].

Likewise, wearables find practical application in habitat monitoring [85]. For instance, observing the behaviors and activities of animals within their natural environment, ensuring the well-being of pets, and studying the flight patterns of birds, among others [86], [87].

Table 2.3 Summary of recent studies in activity recognition domain [1].

Ref.	Aim of study & Major findings	Merits	Demerits
[88] 2019	A framework to co-optimize the operation of sensors and classifiers by dynamically controlling the sampling rate and powering down accelerometer sensors for low-intensity user activities	High accuracy, low power consumption	Not suitable for high-intensity user activities
[89] 2019	An embedded deep CNN multimodal time-series signal classification scheme	Low power, scalable	Complex implementation
[90] 2018	An IoT-based solution for apportioning of the total energy consumption of a household to individual occupants	Accurate, scalable, privacy-preserving	No energy apportioning for heating, ventilation, and air conditioning
[91] 2017	A context-aware framework to offload tasks from wearables to the gateway and cloud	Low latency for interactive user tasks, low energy consumption for tasks unrelated to user interaction	Not tested with a battery-operated smartphone (only with an external power supply)
[92] 2017	An adaptive compressed sensing framework for coarse-grained activity recognition to find an optimal trade-off between compression ratio of each activity type and the overall performance of the activity recognition system through feedback	High accuracy, low power, autonomous feedback system, adaptive activity-specific compressed sensing	Additional processing cost for on-node feedback generation
[93] 2017	A lightweight and low-profile wearable monitoring system for long-term activity monitoring and recognition using two accelerometers instead of a gyroscope as a low power alternative	Low power, high indoor efficiency, accessible to use (wrist-worn)	Dummy data used for processing
[94] 2016	A generalized activity recognition algorithm for implementation in wearables facilitating activity-based communication for the connected industrial worker	Computationally inexpensive, memory-efficient, user-independent	Transition between states is not detected efficiently
[95] 2018	A prototype for emotion recognition system based on low power SoC inside a tiny wearable device using ad hoc simplification	Low complexity, low computational resources	Low accuracy
[96] 2019	A communication network edge maintenance system based on smart wearable technology. It uses a MCSS algorithm for task division and KM for accessing a MEC server	Reduced transmission delay and energy consumption, efficiency of on-site maintenance work	Technological aspects of the used wearable device not provided
[97] 2016	A gesture recognition systems for industrial workers with a ML model using a wrist-worn wearable device	Reduced computational complexity, user-independent	No analysis for the scarce resources availability on the wearable device

Table 2.4 Main parameters considered by recent studies in activity recognition domain [1].

Ref.	Energy Consumption	Accuracy	Latency	Battery Lifetime	Sensitivity
[88]	✔	✔			
[89]	✔	✔			
[90]		✔			
[91]	✔		✔		
[92]	✔	✔			
[93]	✔			✔	
[94]		✔			
[95]		✔			✔
[96]	✔		✔	✔	
[97]		✔			

According to the data presented in table 2.4, the most frequently investigated performance parameters within the activity recognition domain include energy consumption and accuracy. In contrast, metrics such as latency, battery lifetime, and sensitivity received comparatively less frequent attention.

The majority of wearables designed for activity recognition depend on continuous sensing of physical movements utilizing different sensors, generating raw data. These recorded inputs go through complex processing and analysis, involving feature extraction and classification, to precisely identify relevant activities. This demanding task is often accomplished through advanced ML techniques, that require powerful computing resources [98], [99]. Given that wearables are usually small standalone devices with inherent computational limitations, the adoption of various strategies becomes important for energy preservation within the domain of activity recognition. Approaches such as task offloading, energy-efficient hardware design, data compression, and approximate computing emerge as highly effective in this regard.

2.2.3 Smart Environment Applications

Recently, wearables have been extensively used in many IoT applications, particularly in the domain of smart environments that facilitate user-centric automation. These smart environments include a diverse scope, ranging from smart cities and buildings to homes and transportation systems, all geared toward advancing urban development and enhancing overall quality of life [105]. For instance, wearables can play a pivotal role in optimizing heat and electricity management within smart build-

Table 2.5 Summary of recent studies providing solutions for Smart Environments domain [1].

Ref.	Aim of study & Major findings	Merits	Demerits
[100] 2017	A SoC wearable integrating brain signals to control the HVAC system and other home devices (lights, fan) through the voluntary eye blinks	Low power, low complexity, low error rate	Bulky, not easy-to-use in everyday life, limited appliances to control
[101] 2014	A low-power resource-preserving MAC protocol for resource-constrained wearables	Reliable, scalable, low power	No analysis provided for the latency
[102] 2016	An optimization algorithm for cloud-based video crowdsensing using resource-constrained wearables and mobile devices	Higher perceived video quality, energy efficient storage, reduced delivery delay, and higher average throughput	Not suitable for applications requiring high video quality
[103] 2016	A wearable, light-EH-assisted sensing, processing and decision-taking RFID tag for integration with a smart garment	High read range, enhanced functionality, flexible interfacing, diverse low-power sensors	High cost and power consumption of the RFID tag
[104] 2018	A framework for monitoring thermal conditions in a building through the use of wearable solutions, parametric models, and the ML techniques through analyzing specific psychophysical conditions	Detection of internal environmental variables close to users, biometric parameters	Limited factors to assess the thermal comfort

Table 2.6 Main parameters considered by recent studies in Smart Environments domain [1].

Ref.	Energy Consumption	Accuracy	Latency	Throughput	Video quality
[100]		✔			
[101]	✔			✔	
[102]	✔		✔		✔
[103]	✔	✔			
[104]		✔			

ings, thereby contributing to an enhanced user experience while promoting resource conservation [106].

Moreover, an actively pursued research field involving wearables is mobile crowdsensing, where users collaboratively generate substantial volumes of data by collectively sensing and sharing information of mutual interest within smart city contexts [107].

Similarly, wearables can also be utilizedf for controlling household devices within smart homes. For instance, wearables can be used to authorize individual access to shared appliances such as refrigerators, washing machines, or shared living spaces like hostels and student residences. Furthermore, this technology can facilitate the monitoring of appliance usage patterns, ensuring fair distribution of electricity consumption among residents.

Table 2.6 shows that, alongside latency, throughput, and video quality, the parameters most frequently analyzed include energy consumption and accuracy.

2.2.4 General Solutions for Wearable Applications

Wearables are finding application in many novel contexts beyond their traditional use cases. Therefore, there exists a prevailing trend to introduce versatile solutions that can be tailored to address specific applications across various domains.

For instance, Nakhkash et al. [108] conducted a study that presented energy consumption profiles for diverse IoT applications operating on resource-constrained wearables. They advocated the effectiveness of software approximations in optimizing energy usage and performance gains.

Similarly, Golkarifard et al. [109] proposed a code/task offloading strategy applicable to wearables, which leverages the computing capabilities of both cloud and nearby devices. They proposed a versatile task scheduler capable of dynamically categorizing tasks for either local or remote processing.

Zheng et al. [111] outlined software engineering support aimed at empowering application developers to harness shared resources among mobile devices. This approach optimizes overall performance by enabling seamless resource sharing, thus enhancing programmer productivity while reducing energy consumption and execution time.

Furthermore, certain studies advocate for the integration of low-power hardware in future wearable development. These studies provide overall energy consumption

Table 2.7 Summary of recent studies providing general solutions using wearables [1].

Ref.	Aim of study & Major findings	Merits	Demerits
[108] 2019	Leveraging error resiliency of the IoT applications to trade accuracy for performance and energy gains through software approximations at different phases of sensing, computation, and transmission	Analytical modeling and characterization of various IoT applications, real-time hardware evaluation	Can not be generalized for any IoT application
[109] 2018	A unified code offloading system for wearable computing to leverage computation resources of nearby and cloud systems with a reference implementation on Google Glass	Programmer-friendly framework, lightweight offloading, run time task scheduler for the decisions, energy-efficient, fast execution, error recovery support	Not suitable for crowded and large environments such as shopping malls resulting in high failure rate even with lots of nearby devices
[22] 2019	A comparison of performance and energy consumption of various platform boards emulating wearables to investigate a best offloading approach for improved QoS for the IoT applications. It proves that offloading computationally intense tasks to a powerful node improves QoS but not always valid for data-intensive tasks	Multithreading, classification of tasks for improved QoS	Communication overhead in terms of power consumption not considered
[110] 2019	An Android-based application for wearables whose tasks can be partitioned between a wearable, MCC, and Fog computing	Enhanced battery lifetime, access to high computing and storage resources of fog and cloud servers	Additional overhead in terms of communication due to the task offloading
[111] 2018	A software engineering support for application developers to leverage the shared resources between heterogeneous mobile devices	Seamless, reliable, and efficient resource sharing among devices	Significant communication overhead for large packets
[112] 2017	Using wake up radios with the BLE transceivers to minimize the energy consumption due to continuous listening	Low energy consumption	Extra hardware cost, no hardware implementation to verify results
[113] 2017	An adaptive channel connection interval for the BLE devices to improve connectivity and energy consumption	Low energy consumption, improved connectivity	The channel link quality assessment overhead, proprietary BLE controllers do not allow such implementations
[114] 2016	A content agnostic privacy and encryption protocol eliminating the need for asymmetric encryption for wearables	Energy-efficient	Limited threat models
[115] 2016	A power aware multi-hop dynamic source routing mechanism for MANET designed for the BLE-based sensor networks	Enhanced lifetime, high throughput	High latency
[116] 2015	A simple and reliable bidirectional communication protocol for communication between the transmitter module (nRF24) and the BLE devices by using advertisement frames	Improved reliability	Increased latency, additional communications overhead, decreased throughput
[117] 2019	A practical deep learning-based framework for improving the performance of wearables	Improved end-to-end latency, robust to the privacy breach, energy-efficient	Memory overheads not considered, the handheld device limited processing capacity can reduce the performance

Table 2.8 Main parameters considered by recent studies providing general solutions for wearable applications [1].

Ref.	Energy Consumption	Latency	Network Lifetime	Throughput	Execution time	Transmission time	Accuracy
[108]	✔						
[109]	✔				✔		
[22]	✔				✔		
[110]	✔				✔		
[111]	✔	✔					
[112]	✔	✔					
[113]	✔						✔
[114]	✔					✔	
[115]			✔	✔			
[116]				✔			
[117]	✔	✔		✔	✔		

profiles and the resultant energy savings achieved through the utilization of such hardware [112], [116].

Moreover, an increasing trend has been observed involving the proposition of generic ML techniques tailored for wearables targeting diverse applications. For instance, Xu et al. [117] introduced a universal deep learning framework specifically designed for wearables. Their approach aims to enhance both performance and energy efficiency. Xu et al. emphasize that the vast data collection potential of wearables, including user activity, healthcare, fitness tracking, and more, opens up a multitude of application areas suitable for deep learning methodologies.

Within these specific application domains, performance monitoring mainly involves energy consumption analysis, followed by latency, throughput, and execution time as summarized in table 2.8. On the other hand, parameters including network lifetime, transmission time, and accuracy are less frequently analyzed.

2.2.5 Statistical Analysis

This subsection presents the discussion on the statistical data obtained through the SLR.

Figure 2.5 provides a statistical breakdown of the evaluation methods used in the literature to compare the performance of proposed solutions against other state-of-the-art techniques. The methods primarily include performance analysis via simulations, real-time experiments on prototypes, or a combination of both simulation results and validation through real-time prototype experimentation.

Based on the collected statistics, it has been observed that 58% of the studies conduct experiments exclusively on real prototypes. Simulation-based analysis without real-time validation constitute about 23% of the total. Whereas, 19% of the studies offer performance evaluations that contain both simulation-based analysis and subsequent validation through real-time experiments conducted on prototypes.

Table 2.9 lists various wearable devices used in the prototype-oriented papers from an energy efficiency perspective.

It is notable that a substantial number of the works focus on creating research-based prototypes that emulate wearable devices rather than utilizing commercially available wearable devices for experimentation with the intent of improving device efficiency.

Drawing insights from the literature, we present a classification of the energy consumption profile (categorized as low, medium, high) of various wearables.

Majority of devices employed within healthcare and activity recognition domains exhibit a low-to-medium energy consumption profile. This is attributed to their limited data rates and the integration of low-power hardware, to achieve extended operational lifetime. Conversely, a subset of solutions within the smart environment and general-application categories showcase medium-to-high energy consumption profiles. This behavior is primarily attributed to the application demands for high data rates and complex processing demands.

HC –	Healthcare	AR –	Activity Recognition	SE –	Smart Environments	GA –	General Applications
L –	Low		M –	Medium		H –	High

Table 2.9 Examples of wearable devices per application area [1].

App. Area	Ref.	Purpose	Wearability	Wearable devices / Research prototypes	Energy Profile
HC	[53]	Autonomic nervous system activities	Finger worn	A ring sensor	L
HC	[58]	Stroke rehabilitation	Upper arm	Armband	L–M

App. Area	Ref.	Purpose	Wearability	Wearable devices / Research prototypes	Energy Profile
HC	[56]	Heart rate variability monitoring	Wrist worn	Custom made wrist wearable ECG sensor	L
HC	[52]	Cardiovascular healthcare system	Chest worn	Customized SoC based ECG sensing patch	L
HC	[61]	Respiratory rate monitoring	Abdomen	A custom made wearable device with 3-axis accelerometer	L–M
HC	[67]	A configurable bio-signal acquisition device	*NA*	A custom made multi channel device capable of acquiring ECG, EMG, and EEG signals	L–M
HC	[60]	A lossy signal compression technique for biosignals on wearables	Chest and wrist worn	Zephyr BioHarness 3 wearable device	L
HC	[64]	A multi sensor 6LoWPAN-enabled WBAN platform	*NA*	Multiple Zolertia sensor motes connected to a main Cubox device	L
HC	[68]	A web-based motion detection system using wearables	Carried in pocket	Zolertia Z1 motes emulating wearables	M–H
AR	[97]	Gesture recognition system for industrial workers	Wrist worn	A custom-built wearable device containing accelerometer and gyroscope sensors	L–M
AR	[92]	Activity recognition	Torso	A 3D motion tracker	L
AR	[88]	Low intensity activity recognition	Knee worn	A custom prototype using an accelerometer as a motion sensor and stretch sensor	L
AR	[95]	An emotion recognition system	Not specified	A custom built wearable prototype including Photo-PlethysmoGraphy (PPG), Galvanic Skin Response (GSR), and Skin temperature (SK) sensors	L
AR	[94]	Activity recognition for industrial workers using wearables	Sacrum worn	A custom-built wearable device using Bosch's BMI 160 containing accelerometer and gyroscope sensors	L
AR	[93]	A wearable system for long-term activity monitoring and recognition	Wrist worn	An nRF51822 System on Chip (SoC) with two ADXL362 accelerometers	L

App. Area	Ref.	Purpose	Wearability	Wearable devices / Research prototypes	Energy Profile
SE	[103]	A wearable RFID tag for smart garments enabling seamless interaction of wearer with other smart devices	Any on-body garment	A custom made circular patch antenna with integrated sensing, processing, and transceiver hardware	L
SE	[102]	Cloud-based video crowdsensing using wearables	Head mounted	Raspberry Pi with a camera module to emulate a wearable device	M–H
SE	[100]	Using EEG signals to control the HVAC system and other home appliances	Forehead mounted	Custom SoC based wearable EEG sensor	L–M
SE	[104]	Monitoring thermal conditions in buildings through wearables	Wrist worn	Empatica E4 wristband	L–M
GA	[108]	Approximating IoT applications for wearables	*NA*	Raspberry Pi Zero emulating a wearable	L
GA	[109]	Code offloading system for a wearable application to extract text information from the ambient environment	Face worn	Google glass	M–H
GA	[117]	Task offloading for wearables	Wrist worn	Smart watch	L–M
GA	[116]	A two-way communication protocol for wearables	*NA*	An nRF24 SoC emulating a wearable device paired with an iPhone 5s	L
GA	[115]	Multihop routing algorithm for BLE enabled wearables	*NA*	Broadcom combo chipset BCM434X for BLE enabled sensors emulating wearables	L
GA	[113]	Efficient connection maintenance technique for dynamic wireless channels	*NA*	A custom built prototype using Raspberry Pi	L

Moreover, given that a substantial portion of the scientific community depends on studies conducted through simulations, it is imperative for researchers to be aware of the recent trends regarding the selection of simulators within the field. To address this objective, we collected data on various types of network simulators used in the literature along with their corresponding utilization percentages.

The statistical findings are presented in figure 2.6. Notably, MATLAB [118] emerges as the dominant choice, accounting for approximately 50%. Several alternative network simulators also found usage, specifically Network Simulator 2 [119], OPNET [120], OMNeT++ [121], and Castalia [122]. Interestingly, all these listed simulators enjoyed equal popularity share of around 5% within the scientific commu-

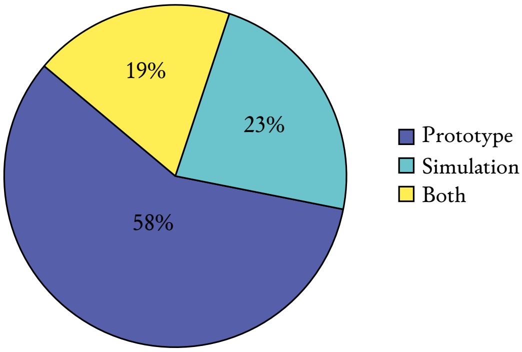

Figure 2.5 Percentage of the evaluation methods used in the analysed literature [1].

nity. A noteworthy observation was made regarding the utilization of custom-built simulators, namely C++ and Java-based, constituting a 10% share. Lastly, approximately 20% of the studies did not mention their choice of simulation tool. The notable lack of uniformity in the choice of simulator emphasizes the potential need for a dedicated simulation tool tailored specifically for the IoWT domain. Such a tool could be purpose-built, widely acknowledged within the community, and instrumental in enhancing result reproducibility.

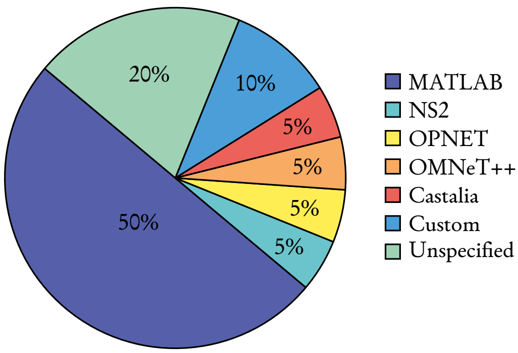

Figure 2.6 Percentage of the network simulators used in the analysed literature [1].

While the selection of communication technology is influenced by the specific demands and limitations of each application domain, it's important to emphasize that the prevailing communication technologies predominantly operate on a short-range.

The gathered data pertaining to wireless technology utilization across various studies is illustrated in figure 2.7. Notably, Bluetooth (including Bluetooth Low Energy (BLE)) emerges as the predominant short-range communication technology within the domain, constituting approximately 46% of the share. Zigbee also finds significant application with a 15% share, followed by Wi-Fi at 9%.

Moreover, several studies suggest a combination of these communication technologies termed as hybrid techniques, accounting for a 9% contribution. The IEEE 802.15.6 standard, commonly known as WBANs, is observed to hold a share of 6%. Lastly, solutions that do not specify the communication technology constitute 15% of the total.

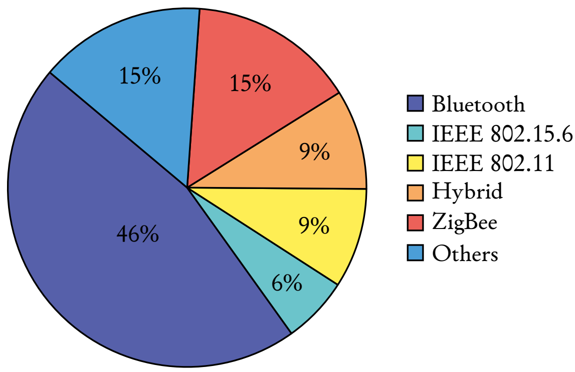

Figure 2.7 Percentage of the communication technologies used in the analysed literature [1].

Finally, the studies were examined based on energy-efficiency focus perspective, as depicted in figure 2.8. From this analysis, it can be concluded that a notable majority of studies address the issue of energy efficiency rather theoretically lacking quantifiable metrics that could be readily converted into energy-related ones. Interestingly, a substantial portion of researchers (29%) have undertaken practical steps, including prototyping and measurements, to validate the efficacy of their proposed energy-efficient strategies.

2.3 Strategies to Improve Energy Efficiency of Wearables

In this section, we provide an extensive discussion on the primary methods utilized in the literature to enhance energy efficiency in wearable devices and related communication networks based on our review.

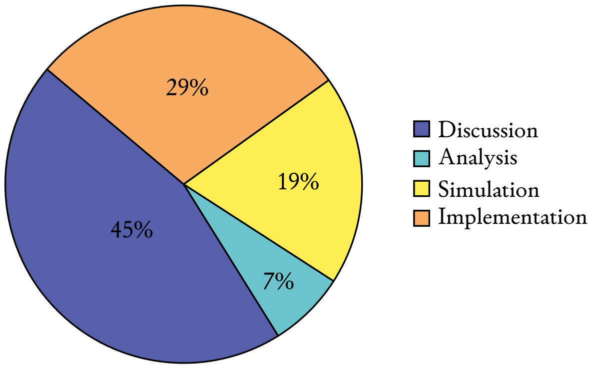

Figure 2.8 Distribution of the energy-efficiency analysis elaboration in the analyzed papers [1].

2.3.1 Task Offloading

Numerous demanding applications require significant computational resources, including tasks such as deep learning methods for face, text, and activity recognition, image and video processing, Augmented Reality (AR)/ Virtual Reality (VR)/Mixed Reality (MR) applications, etc. Such tasks can significantly reduce the battery life of wearable devices. As a result, a prevalent approach in the literature to enhance energy efficiency in wearables involves computational task offloading. This involves transferring resource-intensive tasks to remote execution locations, such as cloud servers, to mitigate processing demands and energy consumption on the mobile device. This strategy has been extensively utilized in the literature [91], [123], [124]. Traditionally, task offloading relied on leveraging cloud services to handle computationally intensive tasks. However, it introduces notable transmission delays that may occasionally fail to meet the stringent latency requirements of many modern IoT applications [125], [126].

Moreover, an additional challenge encountered in task offloading to the cloud, was the absence of a reliable and consistent Internet connection [109]. Modern handheld devices, such as smartphones, now boast more robust chipsets featuring multicore processors, which present a promising alternative. Consequently, the utilization of edge computing [19] and fog computing [127] techniques to leverage the resources of mobile and gateway devices, has demonstrated substantial advantages for wearable devices facing limitations in resources both in terms of energy consumption and performance [128]–[130].

Additionally, more advanced wearable devices come equipped with multiple communication options including low-power short-range communication technologies, such as BLE, as well as Wi-Fi and/or cellular connectivity options. Thus, eliminating the reliance on Internet connectivity to access cloud services. However, a drawback of using low-power communication technologies is low datarate that results in increased latency [131]. Hence, task offloading over BLE could be a feasible option for lightweight tasks only. Whereas, utilizing Wi-Fi can enable a wearable device to create an ad hoc Wi-Fi link with nearby devices, for example, a smartphone, to leverage high computation and energy resources with high data rates even in the absence of Internet connectivity.

Another complexity arises from the need to efficiently partition tasks into segments that can be executed both locally and remotely, operating independently on nearby devices [22]. Hence, there are multiple considerations and challenges that need to be carefully analyzed to enhance the overall performance of a wearable device through task offloading.

2.3.2 Duty Cycling

Wearable devices typically incorporate a computational unit with storage, a communication unit, and various sensors integrated into their design [4]. These components collectively contribute significantly to the overall energy consumption of the wearable device, especially if they remain active at all times. However, there are certain applications where these components are not utilized with high frequency. Examples include scenarios such as long-term environmental monitoring [132], smart agriculture and livestock monitoring [133], and extended healthcare applications [134], [135], among others.

Therefore, duty cycling is an alternative strategy to conserve energy that involves turning off specific hardware modules or transitioning them into a sleep mode when they are not actively being used.

Efficiently determining the duration and timing of sleep cycles is crucial for achieving this optimization. Otherwise, it could have detrimental effects on various aspects of the wearable device's performance, including execution speed, responsiveness, and latency. Some recent studies suggest leveraging AI-driven methods like RL to design intelligent Medium Access Control (MAC) protocols tailored for IoT devices. These protocols aim to predict wakeup schedules effectively and enable adaptive management of sleep cycles to conserve energy [136].

2.3.3 Energy Aware Routing

Wearable devices frequently connect with other wearables or mobile devices in their vicinity to establish communication with a remote instance, which could be either an edge/fog node or a cloud-based infrastructure [137]. This connectivity is particularly relevant in healthcare scenarios, where multiple wearable devices collaborate to monitor a patient's overall health condition. They achieve this by transmitting data to a common data collection point, which subsequently forwards the data to a remote medical facility for in-depth processing and analysis [138]. Therefore, energy-efficient routing protocols are employed to extend the operational lifespan of wearable devices. This strategy helps conserve the energy of network nodes that might otherwise be excessively engaged in relaying data, leading to premature battery depletion [139].

While energy-aware routing appears to be a promising strategy, there are certain associated overheads linked to the determination of optimal routing paths [140]. Nodes must possess knowledge about the remaining energy levels of their neighboring nodes, which necessitates the exchange of periodic control messages to share available resources. Therefore, a careful balancing of trade-offs becomes critical during the design of an energy-aware routing approach.

2.3.4 Low-power Hardware Design

As electronic equipment design progresses, several components supporting low-power computing, communication, and sensing have been developed to enhance the battery life of future wearables [141], [142]. This evolution has given rise to the concept of low-power hardware design. Furthermore, various prototypes have been created using low-power and miniaturized Application-Specific Integrated Circuit (ASIC) hardware design architectures [143]–[145].

In many wireless devices, the communication subunit is commonly identified as the most energy-consuming element [146]. Despite the implementation of duty cycling techniques, a notable amount of energy is still consumed in monitoring the wireless channel for incoming messages and minimizing the risk of collisions with concurrent transmissions [147]. Furthermore, in highly dynamic and crowded scenarios wearable devices are required to continually sense nearby devices for potential data exchanges. Hence, the radio remains in a listening state for extended periods. This situation leaves limited room for effective duty cycling [148].

To address these challenges, certain studies advocate for the integration of supplementary hardware units with near-zero power consumption, known as wakeup radios [149]. These devices primarily serve the purpose of monitoring wireless activity, triggering the main radio unit to activate only when necessary. These wakeup radios demonstrate notable energy efficiency. However, the trade-off lies in the additional cost and space requirements associated with integrating these wakeup radios alongside the primary communication units on wearable devices.

2.3.5 Low-power Communications

Majority of available wearable devices utilize short-range wireless communication technologies, including BLE [150], Zigbee [151], and Wi-Fi [152], among others.

However, the selection of a communication technology depends significantly on the specific application's requirements. For instance, when the intended application demands high data rates, Wi-Fi becomes a suitable choice. Conversely, employing communication protocols with high data rates and power consumption can prove inefficient for many IoWT applications [153].

In general, low-power short-range communication technologies significantly minimize power consumption associated with data transmission. Notably, technologies like BLE and Zigbee have demonstrated enhanced energy efficiency [131], [154], [155]. Furthermore, the emergence of low-power, long-range non-cellular technologies such as LoRa [156], Sigfox [157], and IEEE 802.11ah [158] present promising options for low-power wearable devices. However, the prevalence of cellular technologies, including Long-Term Evolution (LTE) for LTE-M and Narrowband IoT (NB-IoT), remains limited in the context of wearables due to existing industrial gaps, as discussed in [159].

2.3.6 Adaptive Transmission Power Control

Among wearable device tasks, data transmission is often considered as the most energy-intensive operation, with sending a single bit potentially demanding around 1,000 times more energy than a single computation [160]. Wearables incorporate onboard transceiver units that are commonly configured to execute data transmission using a fixed high transmission power. This configuration ensures coverage within a defined area [161]. However, there exist scenarios where achieving effective data communication to a nearby node can be accomplished using comparatively lower transmission power. This can be achieved by dynamically adjusting the transmission power based on the surrounding environmental conditions, employing strategies for adaptive transmission power control.

For applications involving frequent data transmissions, particularly those with high transmission intensity, this issue becomes even more severe. Therefore, consistent fixed high power data transmissions can prove to be highly inefficient. In contrast, an adaptive transmission power control mechanism can significantly enhance energy efficiency [162]–[164]. However, to estimate the required transmission power accurately, the transmitting node needs information about the relative distance to the receiver. In cases where nodes are not stationary, achieving this may necessitate the exchange of periodic control messages among nodes.

Recent studies have also proposed the application of lightweight ML-based intelligent transmission power control schemes. In these schemes, nodes iteratively learn about their remaining energy levels and dynamically adjust their transmission powers. This ensures not only minimal energy consumption but also upholds a designated minimum packet error rate [165].

2.3.7 Compressive Sensing

Compressive sensing stands as a signal acquisition and reconstruction technique that leverages the sparsity of signals to achieve notable efficiencies in energy consumption, bandwidth utilization, and overall performance gains [166], [167]. This method facilitates an optimal reconstruction of the original signal using significantly fewer samples compared to what would be required according to the Nyquist criteria. Multiple studies have highlighted the advantages of applying compressive sensing in optimizing power consumption [167]–[169]. Many wearable applications, including those in healthcare, rely on sparse signals. Therefore, sticking to fixed sensing intervals as dictated by the Nyquist criteria may not yield optimal efficiency. However, it is important to note that compressive sensing is only suitable for sparse signals.

Conversely, numerous other applications still necessitate higher sampling rates to effectively reconstruct the desired signal at the destination. In such cases, some studies advocate for the use of adaptive compressed sensing, particularly in situations where the nature of the generated signal remains uncertain. For instance, various activity recognition applications might end up wasting energy through fixed periodic sampling during periods of inactivity. In such scenarios, adaptive compressed sensing demonstrates its effectiveness by dynamically varying sampling rates to preserve energy as needed [62]. However, the challenge remains in determining the ideal sampling rate in dynamic conditions. Moreover, secure compressive sensing finds application in wireless communications as a cryptosystem, with the measurement matrix serving as a key to secure data exchange between communicating entities [170]. Furthermore, its relevance has also extended to the domain of cognitive radio communication [171].

2.3.8 Data Compression

Generating and processing data is a fundamental task for any wearable device [172]. However, applications such as healthcare and activity recognition heavily rely on data sensing, involving continuous data generation, resulting in substantial data volumes that might be correlated, redundant, or inefficient in specific contexts. Thus, efficient data compression can significantly enhance the overall device performance [173]. Efficient management of generated data along with the elimination of redundant and unnecessary data elements, significantly reduces data size and processing duration while at same time extending the device's battery life [174].

Hence, data compression techniques have been widely used across various studies to reduce the dataset requiring processing and transmission, thereby enhancing energy efficiency in both computation and communication phases [175]–[177]. The majority of proposed data compression algorithms strive to optimize the compression ratio. This ratio reflects the extent to which redundant data is eliminated while maintaining a certain value of Root Mean Square Error (RMSE) and SNR, both of which hold significance within numerous IoWT applications [60].

2.3.9 Approximate Computing

Approximate computing is an approach where calculations prioritize rapid delivery of "good enough" results rather than absolute precision. It has emerged as an effective method to enhance performance and energy efficiency in devices with limited resources such as wearables [178]. Given that numerous applications within the IoWT domain rely on data that is often redundant and inherently noisy, the concept of approximate computing allows for trading off accuracy in favor of energy and performance improvements [179]. In various applications such as ML, signal processing, image processing, and big data analytics, achieving results that are sufficiently accurate rather than perfectly precise, can often fulfill the intended purpose [180].

However, the primary challenges associated with adopting approximate computing include determining the threshold for the necessary minimum accuracy specific to each application, identifying tasks suitable for approximation within the execution flow, and monitoring the outcomes of the application [181]. Therefore, careful fine-tuning of approximation techniques becomes essential to achieve optimal performance enhancements including execution speed, latency, and energy efficiency.

2.3.10 Security Primitives-related Aspects

The majority of current wearable devices depend on conventional information security measures that were not originally designed for the energy efficiency requirements of resource-limited devices. Currently, developers and researchers are dedicating efforts to explore information security solutions tailored for wearable technology particularly crucial for medical and industrial applications.

Comparing the execution time of various cryptographic primitives such as symmetric and asymmetric cryptography, block ciphers, elliptic-curve cryptography, and standard hashing functions, authors in [182] advocate the necessity to design lightweight cryptographic primitives that balance energy consumption and security.

Additionally, the migration of blockchain systems toward wearable devices has been foreseen as an essential progression in the evolution of distributed systems [183], [184]. However, the researchers emphasize the necessity to develop and implement novel consensus mechanisms for wearables to mitigate the adverse impact of cryptographic primitives execution on battery lifetime.

Given that wearables often come as commercially available devices lacking open-source operating systems with a few exceptions [185], the creation and incorporation of innovative energy-efficient security measures are still at an early stage. Whereas, a significant portion of smaller developers tend to overlook considerations related to security and privacy to enhance energy efficiency.

2.4 Summary

To summarize, this chapter presents a SLR of state-of-the-art solutions aiming to improve energy efficiency within the IoWT domain. We proposed a taxonomy of IoWT solutions, considering their energy efficiency perspective and categorizing them based on their targeted application areas. These categories include healthcare, activity recognition, smart environments, and general solutions. Notably, a significant portion of existing solutions primarily focused on healthcare-related applications since wearables were historically developed for specific medical purposes. Nevertheless, in recent times, with advancements in the field, wearables have found applications in diverse domains beyond healthcare. Moreover, we presented a statistical analysis of the available solutions over the years, examining their publication trends. This analysis revealed a continuous increase in wearable-related research, suggesting a growing interest in the field that is anticipated to persist in the coming years.

Furthermore, we provided a comprehensive discussion presenting qualitative and comparative analysis of existing studies within each category, providing insights into their merits, demerits, main performance parameters, and major contributions. Additionally, we presented a statistical analysis, to find out the commonly utilized tools for evaluating the performance of proposed solutions. This analysis revealed a general trend among researchers to develop prototypes to validate the efficiency of their proposed solutions. Nonetheless, some studies only presented simulation-based results, with MATLAB emerging as the most commonly used simulator, among others. In contrast, a portion of the studies presented a combination of simulation-based findings alongside real-time experiments conducted on prototypes. Similarly, another statistical analysis was provided to highlight the most frequently utilized communication technologies in wearables. This analysis showed that BLE was the most commonly used, primarily owing to its low power consumption characteristics, besides others. Finally, to facilitate new researchers in the field, we presented a summarized discussion highlighting the principal techniques found in the literature for enhancing energy efficiency in wearables, emphasizing the challenges/trade-offs associated with these approaches.

3 TASK OFFLOADING FOR WEARABLES IN A TWO-TIER EDGE ARCHITECTURE

Small form-factor electronics, including devices such as smartwatches, smart glasses, wrist bands, AR/VR/Extended Reality (XR) headsets, etc., are becoming increasingly popular these days with an increasing range of applications contributing to this trend [186]. With the technological advancements, wearables are becoming more capable in terms of communication as well as sensing by incorporating several different sensors thus opening avenues for many new applications. Some of these applications involve computation-hungry use-cases such as image or video processing and compression, among others. Historically, wearable and handheld devices were not optimized for executing such demanding tasks due to factors like limited battery capacity and heat generation.

Offloading these computation-intensive tasks to a more powerful and energy-efficient nearby device can significantly enhance the overall user experience as well as conserve the wearable's limited resources.

This chapter examines various scenarios for offloading tasks from wearables to an edge device. The rest of the chapter is organized as follows:

Section 3.1 provides the motivation behind this work including the main contributions.

Section 3.2 presents the system model including reference architecture, main assumptions, and mathematical formulation for the desired performance metrics.

Section 3.3 provides the numerical results and performance evaluation of the proposed model.

Section 3.4 concludes the work with our major findings.

3.1 Motivation

Task offloading refers to the process of communicating input data (for a task originally generated at the wearable device) for processing to a nearby computing entity with more resources before returning the result to the originating device [21].

In addition to dedicated edge servers, modern smartphones come equipped with advanced chipsets housing multicore processors. This design enables these smartphones to effectively function as edge devices, offering users the capability to access a multitude of computation-heavy applications on wearables. Hence, through offloading tasks to the smartphone, wearables can efficiently execute compute-intensive applications such as image processing, employing ML algorithms for face, text, and activity recognition, as well as supporting augmented reality applications without quickly depleting the battery otherwise [123], [124]. Several task offloading solutions for wearables and other mobile devices have been proposed in the literature [91], [109], [117], [124], [187], [188].

Task offloading also enhances the storage and computational capabilities of mobile devices as additional benefits beyond energy conservation. However, the benefits achieved are not usually straightforward and require a case-by-case analysis to determine *when, what, and where* to offload. Notably, there are situations where the energy expended by the wearable in transmitting data to the task executor might exceed the energy used for local computation, making task offloading inefficient. Moreover, certain conditions could lead to an increase in the overall task accomplishment time through offloading, for instance, due to transferring large input data across low-capacity wireless connections [22]. Therefore, it is critical to accurately assess the benefits of task offloading in terms of energy consumption and the extent to which it meets the latency requirements of various applications.

3.1.1 Contributions

The main contributions we provide in this chapter are briefly reiterated as follows:

- We present a two-tier edge architecture for task offloading from wearables to the edge including a smartphone and an edge server.
- We provide a comprehensive analysis of task execution performance on wearable devices, identifying the limits and constraints.

- We explain the scenarios in which task offloading to the edge can enhance performance and to what extent, in terms of task accomplishment time and energy consumption.

3.2 System Model

This section introduces the reference architecture, followed by the main assumptions and mathematical formulation to derive the desired performance metrics.

3.2.1 Reference Architecture and Main Assumptions

We consider a scenario that includes a wearable gadget like Google Glass [189] that is paired with the user's smartphone using a short-range wireless connection which functions as a gateway to the Internet, as depicted in figure 3.1. The smartphone, in turn, establishes a link through a base station (BS) that hosts an edge server. Without loss of generality, we will refer to the glass as the wearable device which needs to execute a computationally intensive task, such as the processing and streaming of AR images and videos. The AR glass can be utilized by the user to capture images/videos while on the move, for various applications like facial recognition and automatic license plate recognition, among others.

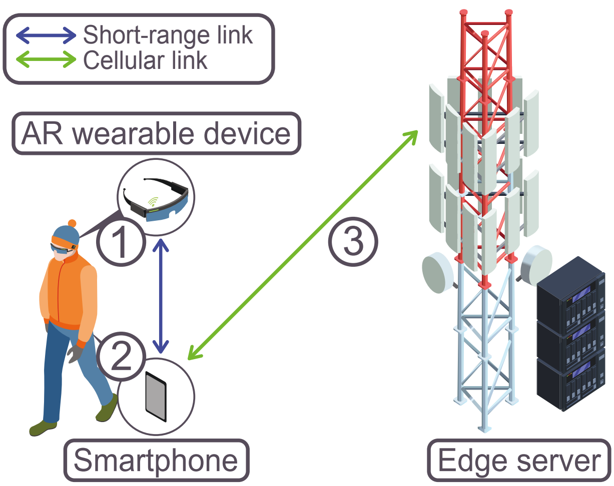

Figure 3.1 Reference architecture and scenarios of interest: (1) – Local execution on wearable; (2) – Offloading to the smartphone; (3) – Offloading to the edge server [190].

The computational task is characterized by the input data, D (measured in bits), and the number of CPU cycles/bit needed to execute the task, denoted as C. An effective approach to approximate C for any task is to utilize a program profiler that observes all program parameters [191]. These program profilers make use of information, including acquired memory, execution time, thread CPU time, the count and type of instructions, as well as function calls, to accomplish this estimation [192].

As shown in figure 3.1, executing any computationally intensive task locally places a significant burden on the wearable device, as indicated in scenario 1. Therefore, our objective is to examine whether offloading such a task from the wearable device to nearby devices such as the smartphone 2 or the edge server 3 could effectively preserve the wearable's energy while meeting the application's latency demands. To this end, we evaluate the performance in terms of energy consumption and task accomplishment time across three distinct scenarios. The wearable device can choose to locally execute the task, which leads to extended task accomplishment time due to its limited computational capacity, thereby degrading the overall user experience. Alternatively, the wearable device can offload the task to either the smartphone or the edge server, both of which possess comparatively much greater computational resources. However, this comes at the cost of additional energy spent by the wearable device to transfer data to the task executor and subsequently receive the processed results.

Our study is based on the following main assumptions:

- Unlike conventional wearables, modern wearable devices often come equipped with a variety of connectivity options including Bluetooth, Bluetooth Low Energy (BLE), Wi-Fi, millimeter Wave, and/or LTE communication interfaces [28]. However, low-power technologies like Bluetooth/BLE are unsuitable for task offloading due to their limited data rates. Consequently, task offloading becomes impractical due to substantial communication delays [193]. Therefore, in an outdoor setting, we assume that the wearable device establishes a connection with the user's smartphone via Wi-Fi, subsequently accessing the edge server through a cellular network.
- Each task under consideration is atomic, meaning it cannot be further divided into smaller subtasks.
- In applications like face recognition, automatic license plate recognition, etc., the output data size is considerably smaller than the input data. Therefore, the

time taken to transfer output data from the task executor (smartphone/edge server) to the wearable device can be reasonably neglected [124], [194].

3.2.2 Mathematical Formulations

We draw upon a theoretical background to compute the relevant metrics for each of the three scenarios: 1 performing the task locally on the wearable device, 2 offloading the task to the smartphone, and 3 transferring the task to the edge server. Specifically, we calculate the task accomplishment time and the energy spent during the entire task execution. A list of main notations used throughout this chapter is provided in table 3.1.

Local task execution on the wearable

The wearable device operates autonomously, performing all computations locally without any offloading.

Task Accomplishment Time

The task accomplishment time for executing a task locally on the wearable device T_w can be estimated as follows [187]:

$$T_w = \frac{D \times C}{F_w}, \tag{3.1}$$

where F_w denotes the processing power available on the wearable device in terms of CPU cycles per second.

Energy Consumption

The CPU power consumption is proportional to the product of CPU frequency F_w and square of supply voltage to the chip, V^2, as given in [195]. Hence, the power consumption can be expressed as:

$$P_w = \alpha(V^2 \times F_w), \tag{3.2}$$

Table 3.1 List of main notations

Notation	Description
C	Computational intensity of a task
CR	Coding rate over a WiFi link
d	Distance between the smartphone and the BS
D	Input data size for a task
DS	Number of data subcarriers over a WiFi link
$E_{d,e}$	Energy consumed at the smartphone for delivering the input data to the edge server
E_e	Total energy consumption in offloading a task for execution at the edge server
$E_{ex,e}$	Energy consumed in executing a task on the edge server
$E_{ex,s}$	Energy consumed in executing a task on the smartphone
$E_{r,s}$	Energy consumed by the smartphone in receiving a task from the wearable
E_s	Total energy consumption in offloading a task for execution at the smartphone
$E_{s,idle}$	Energy consumed by the smartphone in idling while the task gets executed on the edge server
$E_{t,w}$	Energy consumed by the wearable to transmit input data to the smartphone
$E_{w,idle}$	Energy consumed by the wearable in idling while the task gets executed on the smartphone
f_c	Carrier frequency on the cellular link
F_e	Computational capacity of the edge server
F_s	Computational capacity of the smartphone
F_w	Computational capacity of the wearable device
$H_{s,e}$	Channel gain over the cellular link from smartphone to the edge server
M	Modulation order over the WiFi link
N_0	Gaussian noise power over the cellular link
PL	Path loss over the cellular link
$P_{r,s}$	Power consumed by the smartphone in receiving data over Wi-Fi
$P_{s,idle}$	Power consumed by the smartphone during idling while the task gets executed on the edge server
$P_{t,s}$	Power consumed by the smartphone in transmitting input data to the edge server
$P_{t,w}$	Power consumed by the wearable in transmitting input data to the smartphone
P_w	Power consumed by the CPU on the wearable device
$P_{w,idle}$	Power consumed by the wearable during idling while the task gets executed on the smartphone
R_s	Data rate experienced by the smartphone over cellular link
R_w	Data rate experienced by the wearable over a Wi-Fi link
SI	Symbol interval time over Wi-Fi
SS	Number of spatial streams used for transmission over Wi-Fi
$T_{d,e}$	Time consumed in input data delivery to the edge server
$T_{d,s}$	Time consumed in input data delivery to the smartphone
T_e	Total time consumed in offloading a task from the wearable to the edge server
$T_{ex,s}$	Time consumed in executing a task on the smartphone
$T_{ex,e}$	Time consumed in executing a task on the edge server
T_s	Total time consumed in offloading a task for execution at the smartphone
T_w	Time consumed in executing a task on the wearable device
V	Supply voltage to the CPU chip
W_s	Bandwidth over the cellular link
α	Effective switched capacitance for CPU

where α is the effective switched capacitance of each processor, which is related to the chip architecture [196]. Moreover, it has been found that the voltage supply V is approximately linearly proportional to the clock frequency of the CPU [195]. Thus, equation (3.2) can be rewritten as:

$$P_w = \alpha F_w^3. \tag{3.3}$$

Therefore, for an input data size of D bits and the computational intensity of the task C cycles/bit, the energy consumption for executing a task locally on the wearable device, E_w, can be estimated as:

$$E_w = P_w \times T_w = \alpha F_w^2 (D \times C). \tag{3.4}$$

Task offloading to smartphone

Typically, a wearable device is paired with the user's smartphone (as illustrated in figure 3.1, scenario 2), which serves as the nearest available device for task offloading, possessing significantly greater resources than the wearable.

Task Accomplishment Time

The task accomplishment time in offloading a task for execution at the smartphone T_s can be defined as the sum of the time consumed in input data delivery to the smartphone over the Wi-Fi link, $T_{d,s}$, and the task execution delay at the smartphone, $T_{ex,s}$:

$$T_s = T_{d,s} + T_{ex,s}. \tag{3.5}$$

The data rate for the wearable device, R_w, to offload a task for execution at the smartphone over Wi-Fi can be estimated as follows [197]:

$$R_w = \frac{DS * M * CR * SS}{SI}, \tag{3.6}$$

where DS represents the number of data subcarriers that transmit modulated data, M represents the modulation order in terms of the number of bits each data subcarrier can represent, CR represents the coding rate, SS defines the number of spatial streams used, and SI is the symbol interval time. An upper bound of 54Mbps can be achieved for a Wi-Fi (802.11g) link based on the values of the above parameters as mentioned in table 3.2.

Hence, the transmission time, $T_{d,s}$, for offloading a task from the wearable device to the smartphone over the Wi-Fi interface would be:

$$T_{d,s} = \frac{D}{R_w}. \tag{3.7}$$

Similar to equation (4.1), the computation delay for executing a task at the smartphone $T_{ex,s}$ is given as:

$$T_{ex,s} = \frac{D \times C}{F_s}. \tag{3.8}$$

Energy consumption

The overall energy consumption in offloading a task for execution at the smartphone, E_s can be expressed as:

$$E_s = E_{t,w} + E_{r,s} + E_{ex,s} + E_{w,idle}, \tag{3.9}$$

where $E_{t,w}$ is the energy consumed by the wearable to transmit input data to the smartphone as:

$$E_{t,w} = \frac{P_{t,w} \times D}{R_w}. \tag{3.10}$$

The energy consumed by the smartphone to receive input data from the wearable is calculated as:

$$E_{r,s} = \frac{P_{r,s} \times D}{R_w}, \tag{3.11}$$

where $P_{r,s}$ is the instantaneous power spent during reception over Wi-Fi as per the measured values in [198].

The energy consumed in executing the task on the smartphone is given as:

$$E_{ex,s} = \alpha F_s^2 (D \times C). \tag{3.12}$$

Finally, the energy spent at the wearable device during idling, while the task gets executed at the smartphone, can be estimated as:

$$E_{w,idle} = P_{w,idle} \times T_{ex,s}, \tag{3.13}$$

where $P_{w,idle}$ is the power spent by the wearable in idle state.

Task offloading to the edge server

Tasks demanding substantial computation that would consume extensive local resources can be shifted from the wearable device to the edge server, as depicted in figure 3.1, scenario 3. When offloading the task to the edge server, the smartphone functions as an intermediary node, receiving input data from the wearable device and transmitting it to the edge server and vice versa for communicating the results back.

Task Accomplishment Time

The total time consumed in offloading the task from wearable to the edge server can be defined as:

$$T_e = T_{d,s} + T_{d,e} + T_{ex,e}, \tag{3.14}$$

where $T_{d,e}$ is the time taken in offloading the task from the smartphone to the edge server over the cellular network, and $T_{ex,e}$ is the time consumed in executing the task at the edge server. Without loss of generality, we refer to the LTE technology for the cellular network. For the uplink transmission from the smartphone to the edge server, the intracell interference is well mitigated in the LTE network [199]. Therefore, the data rate experienced by the smartphone can be given as [187]:

$$R_s = W_s \log_2\left(1 + \frac{P_{t,s}H_{s,e}}{N_o}\right), \tag{3.15}$$

where W_s gives the user bandwidth, and $P_{t,s}$ denotes the transmission power of the smartphone, $H_{s,e}$ denotes the channel gain from the smartphone to the BS, including path loss and fading, and N_o is the Gaussian noise power in the channel. Channel gain $H_{s,e}$ is the reciprocal of path loss. As per the 3GPP standardization [200], for a general non-line-of-sight (NLOS) case, the path loss (in dB) can be estimated as:

$$PL_{NLOS}(d) = 36.7\log_{10}(d) + 26\log_{10}(f_c) + 22.7, \tag{3.16}$$

where d is the distance between the smartphone and the BS (in meters) and f_c is the carrier frequency (in GHz)[1].

Hence, $T_{d,e}$ can be given as:

$$T_{d,e} = \frac{D}{R_s}, \tag{3.17}$$

and, similarly to equation (4.1), the computation delay for executing the task at the edge T_e can be estimated as:

$$T_{ex,e} = \frac{D \times C}{F_e}, \tag{3.18}$$

where F_e is the computational capacity of the edge server.

Energy consumption

The overall energy consumption in offloading a task for execution at the edge server can be expressed as:

$$E_e = E_{t,w} + E_{r,s} + E_{d,e} + E_{ex,e} + E_{w,idle} + E_{s,idle}, \tag{3.19}$$

where $E_{d,e}$ is the corresponding energy consumption at the smartphone for delivering the input data to the edge server over cellular network and can be expressed as:

$$E_{d,e} = \frac{P_{t,s} \times D}{R_s}. \tag{3.20}$$

The smartphone energy consumed while idling, i.e., when the task is executed at the edge server, is calculated as:

$$E_{s,idle} = P_{s,idle} \times T_{ex,e}. \tag{3.21}$$

The energy consumed at the wearable device (the task is offloaded from the smartphone to the edge server and executed at the edge server) is calculated as:

$$E_{w,idle} = P_{w,idle} \times \left(T_{d,e} + T_{ex,e}\right). \tag{3.22}$$

[1]The estimation in equation (3.16) is applicable for the carrier frequency range of 2–6GHz for different antenna heights with the maximum modeling distance range of 2,000 m between the mobile station and the base station, which suits our considered scenario [200].

Table 3.2 Main system parameters [190].

Parameter	Numerical Value [Ref.]
C	10^3 cycles/bit [201], [202]
CR	3/4 [197]
d	100, 300, 600 m [187]
D	0.2-2 MB [123]
DS	48 [197]
f_c	2.1GHz [203]
F_e	20GHz [187]
F_s	2.2GHz [204]
F_w	1GHz [189]
M	6 [197]
No	-113dBm [202]
$P_{r,s}$	0.94W [198]
$P_{s,idle}$	30mW [205]–[207]
$P_{t,s}$	0.2W [187]
$P_{t,w}$	1.28W [198]
$P_{w,idle}$	22mW [208]
SI	4□s [197]
SS	1 [197]
W_s	1MHz [187]
α	10^{-28} [196]

Finally, the energy consumed in executing the task on the edge server is:

$$E_{ex,e} = \alpha F_e^2 (D \times C). \tag{3.23}$$

However, due to its co-location with the base station and lack of significant energy constraints, in contrast to the other battery-powered devices within our system model, namely, the wearable and the smartphone, the energy consumption associated with receiving input data from the smartphone and subsequently processing it on the edge server is considered negligible.

3.3 Numerical Results and Discussion

In this section, we present different sets of numerical results achieved through MATLAB simulations derived under settings summarized in table 3.2, unless separately stated.

3.3.1 Local Task Execution on the Wearable

Task accomplishment time graph for local execution on the wearable device when no offloading is employed is shown in figure 3.2. The graph presents the effect of varying input data sizes and different computational capabilities of the wearable device, represented by CPU frequencies ranging from 0.5GHz to 1.2GHz. It is observed clearly that higher CPU frequencies result in shorter task accomplishment times. Hence, wearable devices characterized by lower computational capacities are expected to benefit more through task offloading.

Figure 3.3 depicts the associated energy consumption on the wearable device through local computation. Devices featuring higher CPU frequencies can achieve lower task accomplishment times. However, it comes at the cost of high energy consumption. Since energy consumption on any device is directly proportional to the square of its CPU frequency, as follows from equation (3.4).

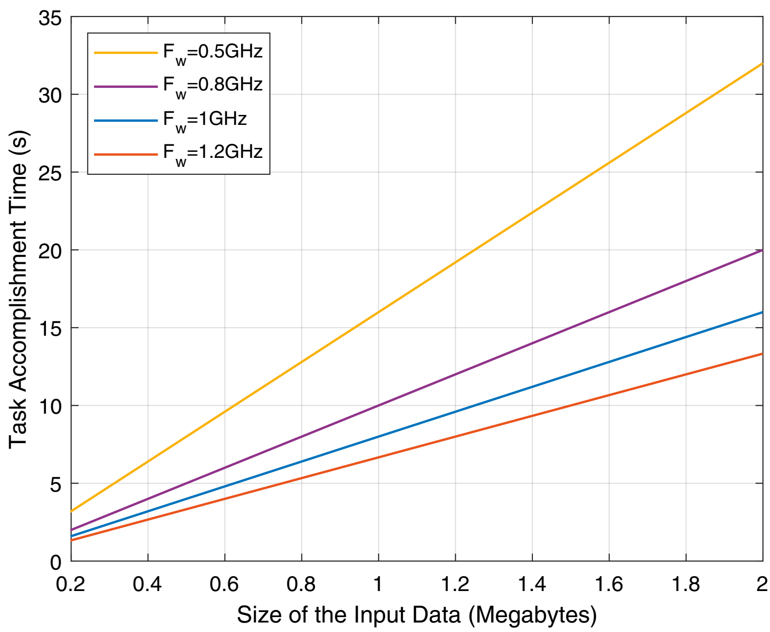

Figure 3.2 Task accomplishment time for local task execution on the wearable with varying CPU frequencies and input data sizes [190].

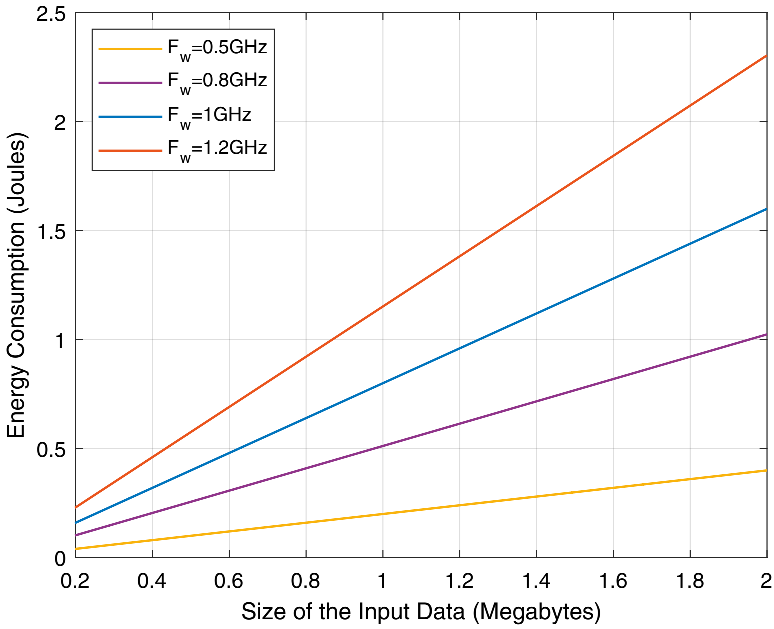

Figure 3.3 Energy consumption for local task execution on the wearable with varying CPU frequencies and input data sizes [190].

3.3.2 Local Task Execution vs Offloading to the Edge

In the subsequent set of results, we show the total time spent during task execution when varying input data sizes. This analysis includes three different scenarios: local execution of the task on the wearable device, offloading to the smartphone (facilitated by a 54Mbps Wi-Fi connection), and offloading to an edge server co-located alongside a base station (BS) at varying distances (100m, 300m, and 600m) from the smartphone (via an LTE connection). The rationale behind considering different distance settings originates from the fact that users carrying both the wearable device and the paired smartphone might be situated in varying locations within the LTE cell. As a result, they could encounter different radio link performances, making these distance settings suitable for evaluation.

As anticipated, figure 3.4 illustrates that as the input data size increases, the task execution time also increases across all scenarios. Local task execution on the wearable device shows the least favorable performance due to its comparatively limited computational resources in contrast to the other scenarios. In contrast, offload-

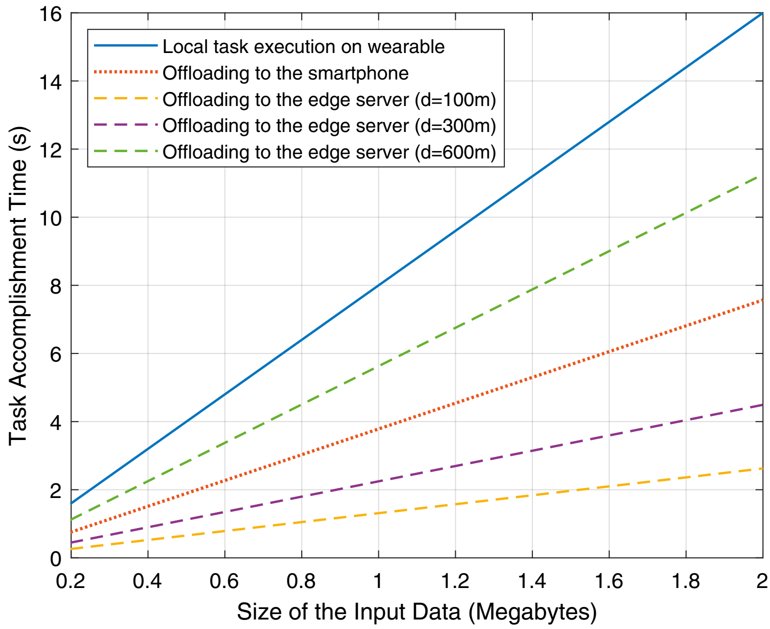

Figure 3.4 Task accomplishment time with varying input data sizes for: *(1)* local task execution at the wearable, *(2)* task offloading to the smartphone, *(3)* task offloading to the edge server (d = 100, 300, 600m) [190].

ing to the edge server when the user is positioned far from the base station (BS) demonstrates the second-worst performance. This can be attributed to the significant degradation in link quality as the smartphone moves away from the BS causing significantly reduced data rates, ultimately resulting in prolonged task execution time. The best performance is observed when offloading to the edge server while the user is situated closer to the LTE BS due to the high data rates achieved in communication as well as the abundant computational resources accessible at the edge server. Offloading to the smartphone produces intermediate performance when compared to the other scenarios. Nonetheless, it is important to consider that variable data rates over Wi-Fi (e.g. 54Mbps in our case) can certainly impact the task accomplishment time. Therefore, offloading time-critical tasks could prove advantageous for fulfilling specific latency requirements while conserving energy on the wearable device.

Figure 3.5 shows the task accomplishment time while varying the distance, d, between the smartphone and the edge server. In this graph, curves for two different input data sizes are depicted, i.e., D = 0.42MB and D = 2MB, corresponding to

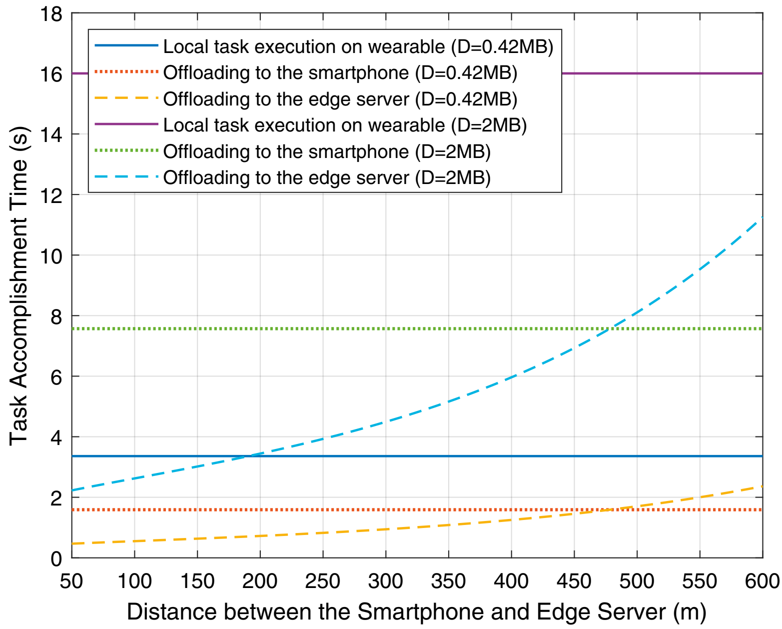

Figure 3.5 Task accomplishment time for two different input data sizes (D=0.42MB and D=2MB) when varying distance between the smartphone and edge server for: *(1)* local task execution at the wearable, *(2)* task offloading to the smartphone, *(3)* task offloading to the edge server [190].

small and large input data, respectively. When offloading tasks to the edge server, the associated cost becomes notably higher as the user moves away from the edge server, particularly for scenarios involving large input data. This increased cost is due to the exchange of large volume of data across both the wireless short-range and long-range links. Additionally, the graph highlights the possibility to achieve task accomplishment times below 1s for smaller input data sizes, particularly when tasks are offloaded to closely situated edge servers.

Furthermore, the breakdown of the overall task accomplishment time for each operational phase can be observed in figure 3.6, encompassing all three task execution scenarios. The transmission time for the smartphone is relatively higher in comparison to the wearable device, attributed to the reduced data rate over a shared cellular network, which becomes more pronounced as the user's distance from the base station increases. Finally, the edge computation time is significantly smaller in comparison to other devices within the system model due to the powerful computa-

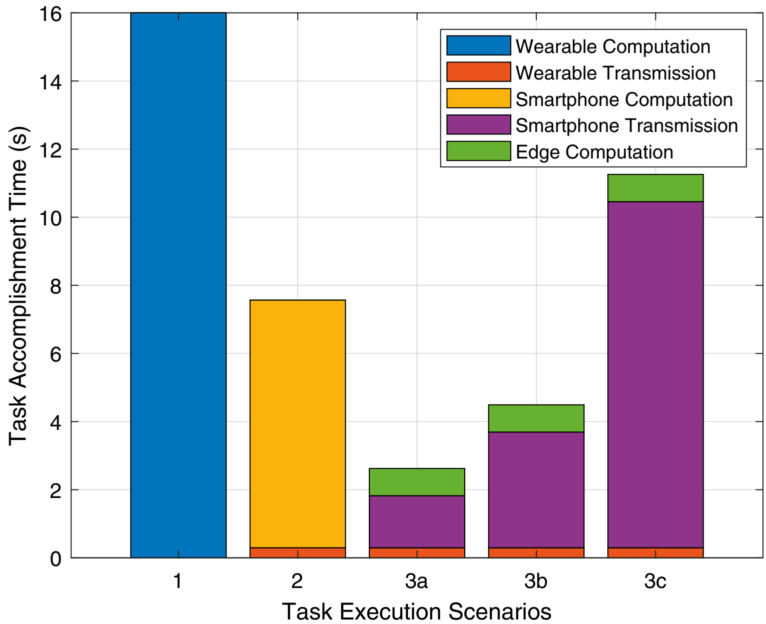

Figure 3.6 Task accomplishment time breakdown for an input data size of 2MB for: *(1)* local task execution at the wearable, *(2)* task offloading to the smartphone, *(3a)* task offloading to the edge server (d=100m), *(3b)* task offloading to the edge server (d=300m), *(3c)* task offloading to the edge server (d=600m) [190].

tional resources available at the edge server.

Similarly, figure 3.7 shows the breakdown of energy consumption for all three task execution scenarios. When considering local execution on the wearable device, the total energy consumption stems solely from the computational processes on the wearable device itself. For the scenario involving task offloading to the smartphone, the collective energy consumption comprises several components. This includes the energy required for the wearable to transmit input data to the smartphone, the energy consumed by the smartphone during the reception of data and task execution, as well as the energy drawn while the wearable remains in an idle state during task execution at the smartphone. Notably, for task offloading to the smartphone, the computational energy consumption is significantly higher than the communication part. Lastly, in the scenario of task offloading to the edge server, the energy consumption comprises multiple segments. This includes energy used by the wearable for transmitting input data to the smartphone, the energy spent by the smartphone

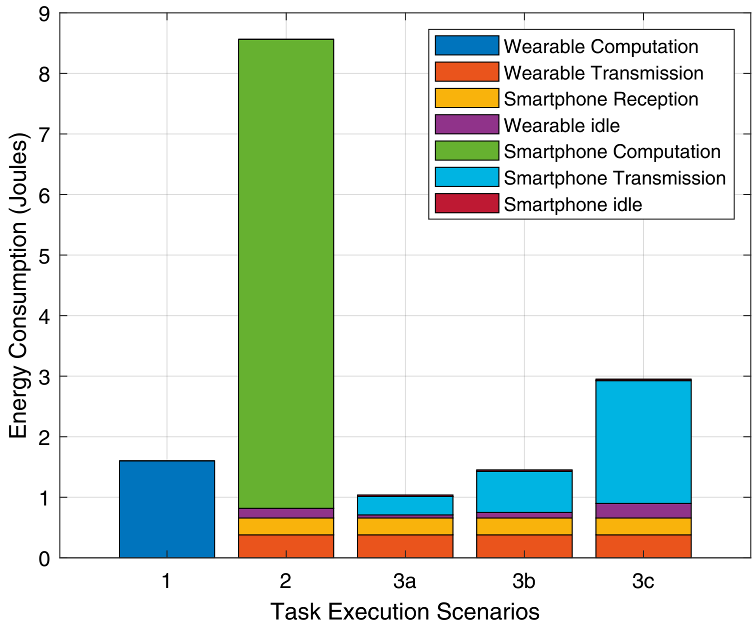

Figure 3.7 Energy consumption breakdown for an input data size of 2MB for: *(1)* local task execution at the wearable, *(2)* task offloading to the smartphone, *(3a)* task offloading to the edge server (d=100m), *(3b)* task offloading to the edge server (d=300m), *(3c)* task offloading to the edge server (d=600m) [190].

for receiving data from the wearable via the short-range link, and energy utilized for transmitting data further to the edge server over the long-range link. In this context, the wearable incurs idle energy consumption until it receives the task output from the smartphone. Comparatively, the energy consumed by the smartphone in idle state while the task is executed on the edge server is much smaller.

3.3.3 Impact of Task Processing Requirements

In addition to input data size, the nature of a task also significantly influences both the overall task accomplishment time and energy consumption.

This effect is shown in figure 3.8, which demonstrates the impact of varying computational intensity on the total task accomplishment time across the three task execution scenarios.

In the case of local execution on the wearable, it can be observed that task execution time experiences a notable increase as computational intensity grows. This

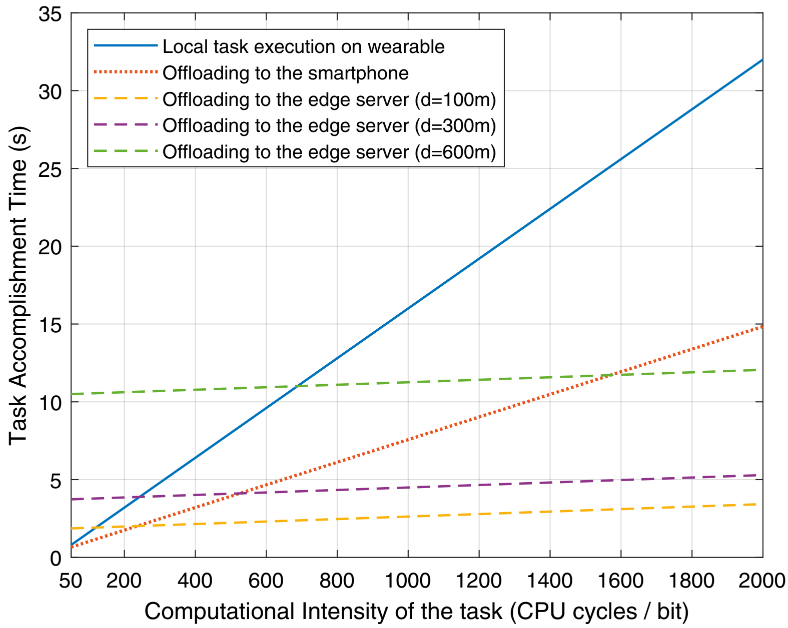

Figure 3.8 Task accomplishment time for an input data size of 2MB with varying task computational intensity for: *(1)* local task execution at the wearable, *(2)* task offloading to the smartphone, *(3)* task offloading to the edge server ($d = 100, 300, 600$m) [190].

effect is due to the intensified processing demands that push the capabilities of the resource-constrained wearable device.

Moreover, a consistent advantageous trend can be seen for the wearable device in offloading tasks to the smartphone as the computational intensity escalates. However, the effectiveness of offloading to the edge server becomes more prominent as computational intensity increases, depending on the user's proximity to the base station (BS). Interestingly, even when a user is positioned far from the BS, offloading to the edge server outperforms offloading to the smartphone for tasks characterized by significant computational intensity, particularly those demanding 1600 CPU cycles/bit or more. This variation is due to the edge server's superior processing capabilities, enabling significantly faster execution times compared to the extended transmission duration over a low-throughput long-range link.

Finally, figure 3.9 depicts the corresponding fluctuations in overall energy consumption for both the wearable device and the smartphone across various computational intensities ($C = 500, 1000, 2000$ CPU cycles/bit) within the context of the

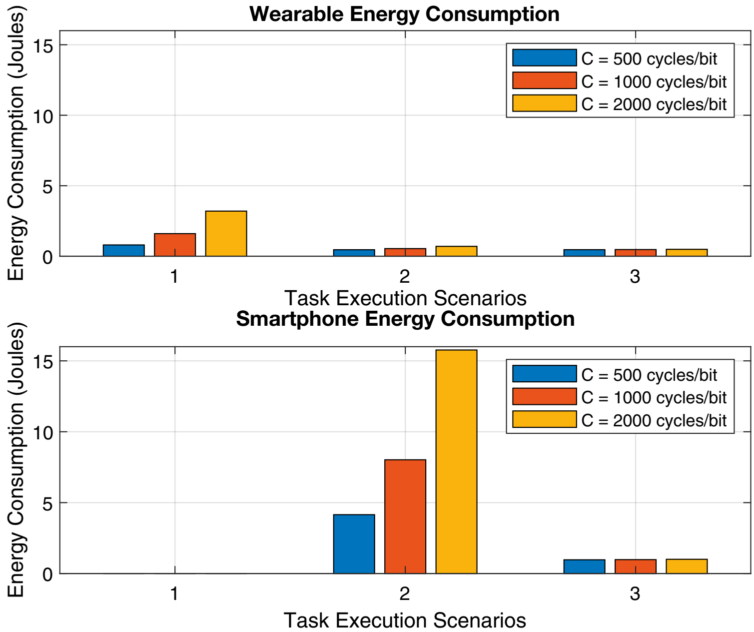

Figure 3.9 Energy consumption per device for an input data size of 2MB with varying task computational intensity for: *(1)* local task execution at the wearable, *(2)* task offloading to the smartphone, *(3)* task offloading to the edge server (d=300m) [190].

three offloading scenarios.

As expected, there is a direct correlation between increasing computational intensity and intensified energy consumption at the devices where the task is executed. Additionally, it is noteworthy to observe that when offloading tasks to the smartphone, the energy spent is significantly higher compared to the energy consumed by the wearable device during local execution of the task. Nonetheless, this is not very concerning, since the smartphones generally have bigger batteries as compared to wearables. Importantly, when the task is offloaded to the edge server, energy consumption values become significantly lower and nearly comparable between the wearable device and the smartphone. This highlights the efficiency of offloading to the edge server, where energy consumption is substantially lowered, especially in comparison to the offloading to the smartphone scenario.

3.4 Summary

In this chapter, we presented a numerical analysis of task offloading for wearables in a two-tier edge architecture that comprises of a smartphone and an edge server as task executors. Our analysis primarily focused on the task accomplishment time and energy consumption parameters.

With the current settings, our results generally demonstrate an advantage in offloading tasks to the edge server over both local execution and offloading to the smartphone; offering benefits such as preserving the wearable's limited energy resources and achieving lower latency in task execution. However, in specific scenarios where the smartphone is situated at the edge of a cell experiencing harsh signal propagation conditions, it is usually better to execute the tasks at the smartphone unless the smartphone is critically low on battery or the task is exceptionally computationally intensive. Furthermore, for light-weight tasks that do not require heavy computation, local execution on the wearable is a preferable approach due to the fact that, in such cases, the time contributed by the wireless transfer of input data can dominate the overall task accomplishment time.

4 REINFORCEMENT LEARNING-BASED TASK OFFLOADING FOR WEARABLES

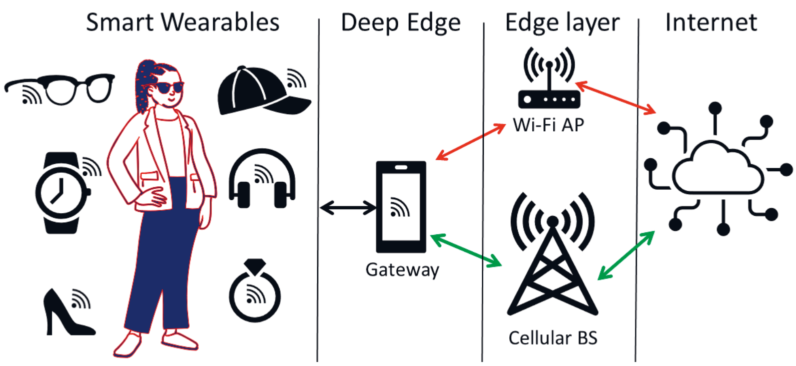

Figure 4.1 The Internet of Wearable Things concept

In the field of computing, task offloading usually refers to the procedure of assigning tasks to some other device in the network that provides computation, storage, and data management services [209]. In the past, Cloud Computing has been extensively used for delivering these services to resource-limited end-user devices. The foundation of cloud infrastructure is usually composed of massive data centers, backbone IP networks, and cellular core networks [210]. Hence, it allowed end users to leverage the extensive computational and storage resources offered by the Cloud.

During the past decades, there has been a remarkable expansion in numerous internet-based companies leveraging cloud services such as Amazon, Dropbox, Facebook, and Youtube, among others. However, the main limitation of cloud computing lies in the considerable time delays faced when accessing the cloud infrastructure via the internet. The latencies supported by cloud computing have proven to be sufficient for many delay-tolerant applications such as social networking, e-commerce, and distance learning. However, cloud computing is unable to guarantee an acceptable level of Quality-of-Service (QoS) for numerous emerging applications that demand ultra low latencies such as interactive online gaming, virtual reality, ultra-high-definition video streaming, automated driving, to name a few [211]–[213]. Additional downsides include the substantial costs and space requirements associated with deploying cloud servers as well as the complicated design and maintenance demands. Moreover, with the proliferation of IoT devices, a multitude of new devices are connecting to the internet, consistently generating substantial data volumes that could potentially congest the core internet infrastructure [214].

More recently, the concept of MEC has emerged as a response to the challenges inherent in offloading tasks to distant cloud servers with the primary goal of bringing cloud services closer to the network edge [20]. Thanks to advancements in processor design, robust edge devices can be deployed at network edges, equipped with substantial processing capabilities and extensive storage capacities. These edge devices essentially function as compact data centers, featuring moderate resources. They are usually co-located with Wi-Fi access points, gateways, and LTE base stations, which are typically set up by telecom operators and internet service providers in close proximity to end-users (ranging from tens to hundreds of meters). Additionally, modern smartphones now come equipped with powerful chipsets featuring multicore processors that can easily handle significantly heavier applications. Since most of the consumer wearable devices commercially available today, pair with the user's smartphone to operate and/or access the Internet. Hence, the smartphone can act both as a gateway node for the wearable device as well as a mobile edge device. Moreover, integrating cutting-edge wireless communication technologies, MEC has the capability to support latencies at levels around tens of milliseconds [215]. Therefore, it has become feasible to execute computationally intensive and latency-critical applications even on low-end devices.

The rest of the chapter is organized as follows:

Section 4.1 provides the motivation behind this work and its significance while also highlighting the main contributions.

Section 4.2 provides the background related to RL-based task offloading citing some state-of-the-art solutions.

In Section 4.3, we present the system model, including the description of reference architecture, main assumptions, and mathematical formulations for local and offloading scenarios.

Section 4.4 presents the Q-learning based task offloading problem formulation.

The details of the proposed task offloading algorithm is presented in section 4.5. Section 4.6 presents the discussion on the performance evaluation of the proposed algorithm.

4.1 Motivation

Delegating tasks to nearby edge devices with enhanced computational capabilities appears as a suitable strategy for resource-constrained devices such as wearables [8]. However, this approach introduces several complexities. Firstly, wearable devices may encounter diverse tasks (with varying attributes including input data size, computational intensity, and desired latency) originating from different applications. As a result, the decision-making process extends beyond the determination of whether to offload or not; it is further complicated by the critical choice of identifying the most suitable task executor node when offloading tasks, e.g., the nearby smartphone or a more powerful edge server. Secondly, the dynamic wireless channel conditions can lead to unexpected degradation in radio link performance due to factors like fading and interference. Consequently, an edge device may not always perform as expected. Hence, devising an optimal offloading policy for an IoT device like a wearable, presents a formidable challenge.

Within this context, the integration of AI into Edge Computing has emerged as a promising solution. Particularly, ML techniques like RL have enabled IoWT devices, such as wearables, to intelligently carry out task offloading by interacting with their environment and employing a trial-and-error approach based on their past experiences, eventually arriving at a nearly optimal task offloading policy [24]. This study analyzes the advantages associated with task offloading for wearables within a dynamic environment. We conceptualize the task offloading problem as a MDP and leverage a model-free Q-learning technique of RL, to allow the wearable device to make optimal task offloading decisions without any prior knowledge.

4.1.1 Contributions

The main contributions we provide in this chapter are briefly reiterated as follows:

- We propose a framework involving a wearable device paired with the user's smartphone, which functions as an edge node for the wearable.

- We provide mathematical formulation for deriving the desired performance metrics, namely, task accomplishment time and energy consumption, in both local computation and offloading scenarios.

- We formulate the task offloading procedure as an MDP and introduce a model-free Q-learning-based algorithm for task offloading.

- We evaluate the performance of the proposed framework in terms of different parameters based on extensive simulations performed in the ns-3 Network Simulator.

4.2 Background

To present a background of the topic, this section presents a discussion on the state-of-the-art in the field of task offloading for wearables and other resource constrained IoT devices.

Identifying the best allocation of tasks to devices is commonly known as the Task Assignment Problem (TAP) in the literature [216]. This class of combinatorial optimization involves making choices about the most suitable location for computing each task in order to minimize factors such as latency and energy consumption.

Addressing this optimization entails the development of efficient solutions approximating optimal results [217]. In the following subsections, we provide an overview of relevant solutions available in the literature, including heuristic-based and RL-based methods to address the TAP in edge computing domain. Moreover, in the final subsection, we discuss how this work aligns with the existing landscape of task offloading solutions.

4.2.1 Heuristics-based Solutions

To find optimal solutions for the TAP, several works have suggested heuristic-based strategies. These heuristics concentrate on identifying sub-optimal but practically feasible solutions. For instance, a Heuristic Offloading Decision Algorithm (HODA) is presented in [187] for joint optimization of resource utilization and the offloading decision in proximate clouds. The authors argue that with the increase in number of users, task completion time and system utility degrades as compared to local execution resulting in a lower Quality-of-Experience (QoE). Therefore, HODA ensures that offloading bottlenecks are not created and prioritizes tasks from mobile users with better utility to enhance per-user performance metrics. However, the downside is that it can only support fewer offloaded users. Similarly, Yang, et al., [91] propose a Context-Aware Task Offloading (CATO) mechanism designed to optimize the allocation of resources for tasks that are offloaded onto smart devices. CATO aims to establish a balance between energy consumption on smart devices and the overall user experience by prioritizing tasks based on contextual information. Consequently, CATO manages to decrease latency for time-critical tasks, while simultaneously conserving energy for other tasks. However, the proposed mechanism is not suitable for traditional solutions that do not include context information.

Moreover, a task offloading application named, Dandelion, has been introduced in [109]. This application enables the offloading of tasks to mobile devices like smartphones, CloudLets, and the Cloud. Tasks are divided into different priority levels and assigned to computing resources using a two-stage task scheduler. The framework effectively achieves task speed up on the mobile device running the Dandelion application and energy efficiency on the device where tasks are offloaded. However, it is important to note that consistently running an application on a resource-constrained device introduces processing and energy consumption overheads, along with a notable increase in heat generation.

4.2.2 RL-based Solutions

More recently, RL has emerged as an innovative approach that offers an optimal or nearly optimal strategy for task offloading. In contrast to the widely recognized supervised and unsupervised ML methods, RL does not rely on pre-existing data sets for offline training. Instead, RL algorithms iteratively learn, making use of instant and past rewards derived from the actions taken within various states. Next, we present some of the most relevant Q-learning based task offloading solutions in the literature.

QLOF [218], is a Q-learning-based computation offloading and resource optimization solution involving mobile device computing, MEC, and Mobile Cloud Computing (MCC). QLOF provides a computation offloading policy that predetermines the computational place for each task from a global perspective. The method also includes an offline optimization mechanism for transmission power and edge cloud computation frequency, to enhance the overall QoE. The performance of QLOF is compared against two other benchmark schemes, namely the all-local scheme and the all-collaborative cloud scheme through MATLAB simulations. The results illustrate that QLOF effectively minimizes the System Loss Function (SYLF) across various scenarios. However, the proposed solution only considers sequential inter dependent tasks originating from a single-chain application.

Authors in [219], propose an Unmanned Aerial Vehicle (UAV)-assisted MEC system for IoT devices. The concept suggests a mobile UAV delivering computing services to resource-constrained IoT devices within the system. The paper introduces a Q-learning-based algorithm designed to minimize the overall system delay, including flying delay, transmission delay, local computing delay, and UAV-based edge computing delay. The algorithm utilizes a time-based reward function for the UAV for both successful and unsuccessful events in serving an IoT device. Using a Python implementation, simulation results reveal that the proposed method effectively reduces the total system delay compared to several benchmark algorithms. However, the proposed scheme only aims to minimize total system delay without considering the energy consumption tradeoff.

Another task offloading and resource allocation solution within the context of the Internet of Vehicles (IoV) is proposed in [220], with an aim to minimize overall system latency and energy usage. The approach employs a Bayesian classifier to categorize tasks based on their specifications related to latency and energy consumption. This classification process dictates the choice of task offloading mode, either Vehicle-to-Vehicle (V2V) or MEC offloading. When a task demands higher energy efficiency, the preference is for V2V offloading; otherwise, the task is offloaded to the MEC server. A Q-learning algorithm is introduced to achieve optimal resource allocation, thereby minimizing the total cost for each offloading mode. Simulation results show that the proposed approach effectively reduces the total system cost. However, it requires extensive training data samples for different offloading modes to generate a Bayesian classifier.

Leveraging the Q-learning technique of RL, a task offloading scheme designed specifically for IoT devices equipped with EH capabilities, is presented in [221]. The main objective of this scheme is to attain an optimal offloading policy without knowing the MEC model, computation latency, and energy consumption model. The proposed approach enables IoT devices to make offloading decisions by considering multiple factors like the current battery level, past radio transmission rate, and a projected EH model. The IoT device evaluates each action by calculating a reward based on a combination of overall latency, energy consumption, task drop loss, and data sharing gains which updates the Q-function iteratively. Furthermore, to expedite the learning process, a transfer learning technique is used. Additionally, a Deep Reinforcement Learning (DRL)-based scheme is also used to compress the large state space, thereby accelerating the learning process. The performance of the proposed schemes is evaluated through simulations, focusing on metrics such as energy consumption, computation latency, and task drop rates. However, the proposed scheme requires an offline training to initialize the Q-values based on the offloading experience in similar environments, i.e., a number of indoor or outdoor networks with similar MEC deployments given the same EH source.

A binary offloading approach based on Q-learning, is introduced in [222], specifically tailored for an application structured as a sequential series of subtasks. The authors argue that conventional Q-learning methods have a drawback concerning the delayed generation of rewards upon completing an action. To address this, the study proposes an experience replay buffer strategy, which temporarily populates the Q-value for a given action immediately. Subsequently, a reward rewind match strategy is employed, replacing the entry with the actual Q-value based on the received reward. To achieve this, the reward function is defined as the negative of the task completion time. As a result, reduced task completion times yield higher rewards, and vice versa. Simulation-based results demonstrate the performance improvements gained from the customized Q-learning algorithm in terms of task completion time. However, the energy consumption tradeoff is not considered.

A Q-learning-based approach for task offloading in a multi-user MEC system is introduced in [223] for joint optimization of offloading decisions and resource allocation. The goal is to minimize the energy consumption of all user devices, taking into account both delay requirements and dynamic resource demands. To achieve this objective, the authors define a reward function in which the reward value is inversely linked to the total energy consumption. Moreover, a mixed-integer nonlinear problem is formulated employing a MDP to establish the connection between offloading and resource allocation policies within the system environment. Numerical results comparing the performance of their proposed method against other baseline techniques are presented. However, only energy consumption related analysis is provided without incorporating other performance parameters particularly the task completion time.

In situations where the agent's states are not finite, the effectiveness of RL solutions reduces, as exploring all possible states to learn action outcomes can be a lengthy process. Consequently, the algorithm's convergence time becomes infinite. To address this issue, researchers have proposed the utilization of Neural Networks to enhance the efficiency of the agent's learning phase. Within this context, a framework called DeepWear, is introduced in [117] tailored for wearable devices, aiming to optimize the offloading of deep learning computations to corresponding handheld devices. DeepWear maximizes the processing capabilities of nearby handheld devices through strategies like strategic model partitioning, pipelining support, and context-aware offloading. Results show that DeepWear is resilient to security breaches and efficientlty balances the trade-off between end-to-end latency and energy consumption on both wearable and handheld devices. However, neural networks are out of the scope of this work.

4.2.3 Our Contributions

Hence, a range of task offloading strategies exist, each with its own advantages and limitations. Notably, AI-driven methods show significant potential. Among these, RL stands out due to its simplicity and intrinsic ability to learn autonomously through reward mechanisms. This chapter introduces the utilization of Q-learning algorithm of RL to enable task offloading from wearable devices to a nearby edge device, such as a smartphone, in the context of IoWT.

Similarly to other works, e.g., [220], [221], we focus on the optimization of two conflicting objectives, i.e., minimizing the task accomplishment time and minimizing the energy consumption. In particular, since we consider the smartphone as an edge node for the wearable to which a task can be offloaded, the energy consumption is the sum of the contributions at the wearable and at the smartphone. Moreover, we propose a more precise energy model as compared to other solutions [91], [109], [220], [221], [223], which also considers idle energy consumption at the wearable device, while waiting for the execution of the task offloaded to the smartphone, and its significance is justified through simulation results.

We develop our proposed framework and carry out its performance evaluation through extensive network simulations performed in the ns-3 simulator that utilizes realistic communication models for various networking protocols and standards, both at the access layer and at the transport layer. The simulations are carried out for multiple applications under a wide variety of realistic settings while also showing how varying the main system parameters of the Q-learning algorithm affects overall performance.

4.3 System Model

In this section, we outline the system model including system architecture, task attributes, main assumptions, and the mathematical formulation for the desired performance metrics.

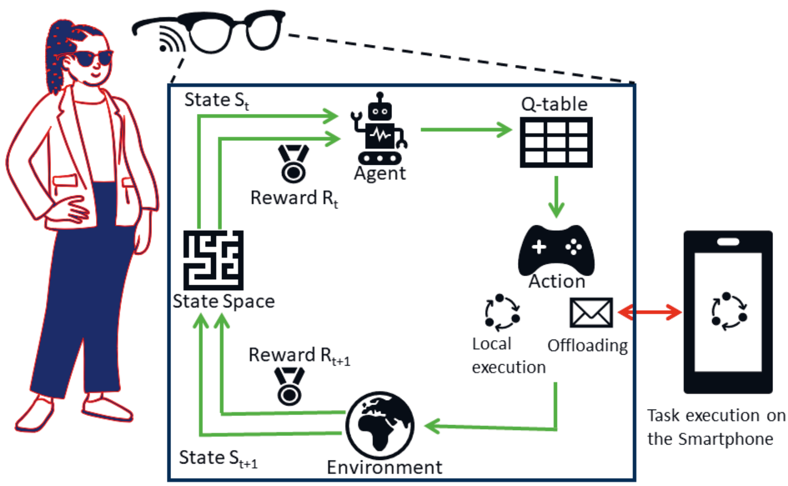

Figure 4.2 System architecture and the Q-learning-based task offloading process

4.3.1 System Architecture

We consider a hierarchical architecture that includes a wearable device, such as an AR headset or smart glasses like Google Glass, worn by a user paired with the user's smartphone through an IEEE 802.11ac (Wi-Fi) wireless link as illustrated in figure 4.2. For the sake of simplification, we will refer to the AR headset or Google Glass as the wearable device that needs to execute computationally-intensive tasks. To meet the application's latency requirements and subsequently enhance the user experience, these tasks must be executed in a timely manner. The wearable device has the option to process these tasks either locally or to offload them for execution to a nearby edge device such as the smartphone which can offer comparably higher computational resources for task execution.

4.3.2 Task Attributes

The wearable device generates computation tasks to be executed. Each individual task, denoted as i, where $i \in \mathcal{I}$, can be described by two main parameters. Specifically, these include the input data size D_i (measured in bits) and the computational intensity of the task, denoted by C_i, representing the number of CPU cycles required per bit for task processing. To accurately estimate C_i, program profilers are used that utilize information like the count and type of instructions, function calls, memory usage, and CPU time allocated to threads [192]. A list of commonly used notations through out this chapter can be found in table 4.1.

4.3.3 Main Assumptions

We make several assumptions for this work. For example, a MEC server (either co-located with a Wi-Fi access point or with a cellular network base station) might not always be available for task offloading, whereas, it is mostly common that a wearable device can connect to the user's smartphone whether in an indoor or outdoor environment. Therefore, we assume that the wearable device connects to the user's smartphone over Wi-Fi. Moreover, we assume that a single task can not be further divided into subtasks. Also, a single task can only be executed at a single device in the network. Furthermore, the output data size is considered to be much smaller than the input data in many applications such as face recognition, automatic license plate recognition, etc. Therefore, the time for transferring output data from the smartphone to the wearable can be safely neglected [124], [194].

In the following subsections, we provide a mathematical formulation for the desired performance metrics for both local and offloading scenarios.

Table 4.1 List of main notations

Notation	Description
$a_{i,s}$	Selected action for executing task i in current state (s)
$a'_{i,s}$	Action generating maximum reward in the next state ($s+1$)
A	Action space
C_i	Computational intensity of task i
D_i	Input data size for task i
$E_{i,idleWoS}$	Idle energy consumption at the wearable while task i gets offloaded to smartphone
$E_{i,exS}$	Energy consumption for executing task i on the smartphone
$E_{i,exW}$	Energy consumption for executing task i on the wearable device
$E_{i,oS}$	Total energy consumption for offloading task i to the smartphone
$E_{i,rxS}$	Energy consumed by the smartphone to receive input data for task i from the wearable
$E_{i,txW}$	Energy consumed by the wearable to transmit input data for task i to the smartphone
F_W	Computational capacity of the wearable device
F_S	Computational capacity of the smartphone
i	Task index
$\mathcal{I}$	Set of all computation tasks in the system
m	Device selected to execute a task
$P_{i,idleW}$	Instantaneous idle power consumption at the wearable
$P_{i,rxS}$	Reception power of smartphone over Wi-Fi
$P_{i,txW}$	Wearable transmission power over Wi-Fi
$Q(s, a_{i,s})$	Q-value for a possible state-action combination
$R(s, a_{i,s})$	Reward for a possible state-action combination
$R_{i,W}$	Data rate experienced by the wearable device over Wi-Fi
s	State of the system
S	State space
$T_{i,dS}$	Time consumed in input data delivery to the smartphone
$T_{i,exS}$	Task execution delay at the smartphone
$T_{i,exW}$	Task accomplishment time for executing task i on the wearable device
$T_{i,oS}$	Task accomplishment time for offloading task i to the smartphone
α	Learning rate
α_c	Effective switched capacitance constant
β_E	Energy coefficient in the cost function
β_T	Time coefficient in the cost function
γ	Discount factor
π	Q-learning policy
π^*	Optimal Q-learning policy
$\psi(s, a_{i,s})$	Cost associated with an action $a_{i,s}$ in state s
$\psi_\pi(\mathcal{I}, \mathcal{S})$	System's long term cost for all the tasks

4.3.4 Local Computing

The task accomplishment time for executing a task i locally on the wearable device, $T_{i,exW}$, can be estimated as follows [187]:[1]

$$T_{i,exW} = \frac{D_i \times C_i}{F_W}, \tag{4.1}$$

where F_W denotes the processing power available on the wearable device in terms of CPU cycles per second also referred to as the computational capacity of a device.

Whereas, energy consumption for executing a task locally on the wearable device, $E_{i,exW}$, can be estimated as:

$$E_{i,exW} = \alpha_c F_W^2 (D_i \times C_i), \tag{4.2}$$

where α_c is the effective switched capacitance of each processor, which is related to the chip architecture [196].

4.3.5 Task offloading to the smartphone

In the following, we provide a mathematical estimation for task accomplishment time and energy consumption for offloading tasks to the smartphone.

The total time consumed in offloading a task for execution at the smartphone, $T_{i,oS}$, can be defined as the sum of the time consumed in input data delivery to the smartphone over the Wi-Fi link, $T_{i,ds}$, and the task execution delay at the smartphone, $T_{i,exS}$:

$$T_{i,oS} = T_{i,dS} + T_{i,exS}. \tag{4.3}$$

The transmission time, $T_{i,dS}$, for offloading a task from the wearable device to the smartphone over the Wi-Fi interface can be estimated as:

$$T_{i,dS} = \frac{D_i}{R_{i,W}}. \tag{4.4}$$

where $R_{i,W}$ is the data rate experienced by the wearable device, to offload task i for execution at the smartphone over Wi-Fi.

[1] Equations 4.1–4.10 are similar to the mathematical formulations presented in 3.2.2. However, they are reproduced here to make the chapter self-explanatory.

Similar to equation (4.1), the computation delay for executing a task at the smartphone $T_{i,exS}$ is given as:

$$T_{i,exS} = \frac{D_i \times C_i}{F_S}. \tag{4.5}$$

The overall energy consumption in offloading a task for execution at the smartphone, $E_{i,oS}$ can be expressed as:

$$E_{i,oS} = E_{i,txW} + E_{i,rxS} + E_{i,exS} + E_{i,idleWoS}, \tag{4.6}$$

where $E_{i,txW}$ is the energy consumed by the wearable to transmit input data to the smartphone as:

$$E_{i,txW} = \frac{P_{i,txW} \times D_i}{R_{i,W}}. \tag{4.7}$$

The energy consumed by the smartphone to receive input data from the wearable is calculated as:

$$E_{i,rxS} = \frac{P_{i,rxS} \times D_i}{R_{i,W}}, \tag{4.8}$$

where $P_{i,rxS}$ is the instantaneous power spent during reception over Wi-Fi.

The energy consumed in executing the task on the smartphone is given as:

$$E_{i,exS} = \alpha_c F_S^2 (D_i \times C_i). \tag{4.9}$$

Finally, the energy spent at the wearable device during idling, while the task gets executed at the smartphone, can be estimated as:

$$E_{i,idleWoS} = P_{i,idleW} \times T_{i,exS}. \tag{4.10}$$

where $P_{i,idleW}$ is the power spent by the wearable in idle state while task i gets executed on the smartphone. To the best of our knowledge, most of the existing works do not consider this power consumption while measuring the total energy consumption on the wearable device. However, it is essential to factor in the idle energy consumption since it is a non-negligible entity and may impact the overall energy consumption significantly, particularly for computationally intensive applications.

4.4 Problem Formulation

In this section, we formulate the process of task offloading in the IoWT as a RL problem [224]. The primary objective is to reduce the overall system cost which is composed of energy and time costs associated with the execution of each task.

MDP serves as a mathematical framework to model decision-making scenarios where an agent interacts with the environment over a sequence of discrete time steps to make informed decisions, while generating an instant reward [225]. Within our considered scenario, task offloading is a sequential decision process in which the wearable device is required to decide the computation side for each task in sequence. Moreover, the state of the system changes as we transition from one time slot to the next. A time slot, in this context, denotes the interval in which a task is computed. Hence, the entire offloading process can be effectively modeled as an MDP.

In this context, Q-learning has emerged as the most widely utilized value iteration technique for offering nearly optimal solutions for extensive MDPs through trial and error without requiring a complete model of the environment [225]. This makes it applicable to real-world scenarios where system dynamics are complex or unknown. Moreover, the iterative approach and use of Q-values allow it to adaptively update its strategy over time, making it effective for finding optimal policies in dynamic environments.

4.4.1 Components of a Q-learning-based Solution

The key components of a Q-learning-based solution (as also illustrated in figure 4.2) are as follows:

Agent

In RL, the device typically responsible for decision-making, is commonly referred to as the agent. In our system, the wearable device generates tasks and must efficiently execute them for the end user. Hence, the wearable device serves as the agent, utilizing Q-learning to iteratively interact with the environment and establish an optimal task offloading policy.

State Space

The system's state is characterized as $s = (i, m)$, where i represents the i^{th} task in the set of all tasks $\mathcal{I}$ and m indicates the device assigned to execute task i. Consequently, the state space is a set of states corresponding to all the tasks within the system, denoted as $S = \{s = (i, m)\}$.

Action Space

The wearable device has the option to decide between executing a task locally or offloading it to the edge device, i.e., the smartphone. Therefore, for each task i, within a given state s, the choice regarding its execution is represented by the variable $a_{i,s}$. Where $a_{i,s} = 0$, represents a local computation action on the wearable device, while $a_{i,s} = 1$, indicates an offloading decision. Consequently, the action space contains the complete set of actions for all tasks within the system and is denoted as A.

Cost Function

As previously mentioned, the wearable device has the option to perform a task locally ($a_{i,s} = 0$) or to offload it for processing on the smartphone ($a_{i,s} = 1$). In either case, a specific cost is associated with each action. This cost typically reflects a trade-off between execution time and energy consumption, a concept that has been widely explored in the literature [226]–[228]. Therefore, we express the immediate cost incurred when taking action $a_{i,s}$ for task i in state s as a combination of energy cost and time cost. Local computation cost is given as:

$$\psi(s, a) = \beta_E \left(\frac{E_{i,exW}}{\sum\limits_{i \in \mathcal{I}, s \in \mathcal{S}} E_{i,ex,W}} \right) + \beta_T \left(\frac{T_{i,exW}}{\sum\limits_{i \in \mathcal{I}, s \in \mathcal{S}} T_{i,ex,W}} \right) \text{ } if \text{ } a_{i,s} = 0, \tag{4.11}$$

whereas, the cost associated with an offloaded task is given as:

$$\psi(s, a) = \beta_E \left(\frac{E_{i,oS}}{\sum\limits_{i \in \mathcal{I}, s \in \mathcal{S}} E_{i,ex,W}} \right) + \beta_T \left(\frac{T_{i,oS}}{\sum\limits_{i \in \mathcal{I}, s \in \mathcal{S}} T_{i,ex,W}} \right) \text{ } if \text{ } a_{i,s} = 1, \tag{4.12}$$

where β_E and β_T represent energy and time coefficients, respectively. It is important to note that these coefficients adhere to the constraint $\beta_E + \beta_T = 1$. Both the energy and time expended during the execution of the current task are normalized with respect to the total energy and time required if all tasks were performed locally on the wearable device.

Consequently, the system's long-term cost $\psi_\pi(\mathcal{I}, \mathcal{S})$, can be defined as the sum of immediate costs for all tasks, as follows:

$$\psi_\pi(\mathcal{I}, \mathcal{S}) = \sum_{i \in \mathcal{I}, s \in \mathcal{S}} \psi(s, a_{i,s}) \tag{4.13}$$

Policy

The policy, denoted as π, is a mapping function that associates each state with a corresponding action within the system. The agent consistently aims to optimize the policy π in order to minimize the long-term cost. Therefore, the optimal policy, denoted as π^*, is the one that minimizes the overall system cost, as expressed by the following equation:

$$\pi^* = argmin_\pi \quad \psi_\pi(\mathcal{I}, \mathcal{S}) \tag{4.14}$$

4.4.2 Q-learning Problem Formulation

Managing available resources in MEC networks becomes extremely crucial, particularly when dealing with battery-powered mobile devices. The primary objective is to find a task offloading strategy that reduces the overall energy consumption on battery-powered mobile devices, including both the wearable and the smartphone, while simultaneously reducing the execution time for all tasks. Consequently, the objective function can be defined as the minimization of system's long-term cost (defined in equation 4.13), expressed as follows:

$$Minimize \;\; \psi_\pi(\mathcal{I}, \mathcal{S}), \tag{4.15}$$

Subject to:

$$a_{i,s} \in \{0, 1\}, \forall i \in \mathcal{I}, \forall s \in \mathcal{S}. \tag{4.16}$$

The constraint outlined in equation (4.16) guarantees that each task can only be allocated to a single available CPU within the system. In other words, each task must be either executing locally on the wearable device ($a_{i,s} = 0$) or offloaded for execution on the smartphone ($a_{i,s} = 1$). Therefore, the variable $a_{i,s}$, is an integer variable that must satisfy the constraint specified in equation (4.16).

4.5 Proposed Q-learning-based Framework

In this section, we present the details of the proposed task offloading framework, which utilizes the Q-learning algorithm of RL to address the above optimization problem.

Q-learning is a kind of iterative RL technique, where the agent learns which actions to take in various circumstances. It is especially well-suited for dealing with large MDPs to arrive at a nearly optimal solution. We employ the model-free Q-learning approach, which doesn't necessitate knowledge of state transition probabilities in advance. In Q-learning-based strategies, while following a policy denoted as π, the agent's objective is to learn the Q-function, denoted as $Q(s, a_{i,s})$, for each possible state-action combination. Consequently, a policy wherein the agent chooses the action with the lowest Q-value for each state, is referred to as an optimal policy.

Every time the system transits from one state to the next, there's an update to the Q-function. Consequently, a distinct Q-value is generated and stored for each state-action pair. The transition to the next state ($s + 1$) occurs when the wearable device effectively executes a task, either locally or through offloading to the edge device in state s. Based on this offloading experience, the wearable device updates the Q-function associated with the state-action pair $(s, a_{i,s})$ using the well-known Bellman's equation [229] as described below:

$$Q\left(s, a_{i,s}\right) = (1 - \alpha)\, Q\left(s, a_{i,s}\right) + \alpha \left(R\left(s, a_{i,s}\right) + \gamma \max_{a'_{i,s} \in A} Q\left(s, a'_{i,s}\right)\right), \tag{4.17}$$

where s represents the current state, while $a_{i,s}$ denotes the action taken in that state. The state $s + 1$ corresponds to the subsequent state following the execution of action $a_{i,s}$ in state s. The parameter $a'_{i,s}$ represents the action that generates the maximum reward achievable in state $s+1$. The learning rate, denoted as α, satisfies the inequality $0 \le \alpha \le 1$. It serves as a parameter that gauges the influence of both past and present

Algorithm 1: Q-learning based task offloading algorithm

Input: State space S, Action space A, Learning rate α, Discount factor γ, and the Exploration probability ε

Output: Q-values for each state-action pair

Initialize $Q(s, a_{i,s})$, $R(s, a_{i,s})$
for *each task* **do**
  Select an action $a_{i,s}$ via ε-greedy policy
  Perform the selected action $a_{i,s}$ by executing the task locally or offloading the input data to the selected edge device
  Evaluate the energy consumption and latency upon task completion
  Evaluate the reward R
  Update Q-table using the Q-function
end
Updated Q-table with near-optimal Q-values for every state-action pair

learning results. A smaller value emphasizes prior learning, while a larger value places more emphasis on current learning outcomes. The discount factor, represented as γ, also satisfies the inequality $0 \leq \gamma \leq 1$. This factor characterizes the impact of future rewards on the current state. A smaller γ value directs the system's focus toward short-term rewards, while a larger γ value emphasizes long-term rewards.

4.5.1 Q-learning Algorithm for Task Offloading

Pseudo Algorithm 1 presents the procedural steps of our proposed methodology. This algorithm requires several inputs, including the state space denoted as S, the action space represented as A, the learning rate labeled as α, the discount factor denoted as γ, and the exploration probability denoted as ε. The exploration probability guides the exploration of new states. The desired output is to get an optimal task offloading that identifies the best action that produces the highest Q-value for each state within the state space.

Hereafter, we detail the specifics of each step within the algorithm. In our proposed algorithm, when the system visits a new state s, it takes an action $a_{i,s}$. Subsequently, the Q-function gets updated, resulting in a new Q-value for that particular state-action pairing, denoted as $Q(s, a_{i,s})$. Initially, Q-values for all state-action pairs can be initialized to zero, given that the system lacks prior learning experiences, meaning that no states have been explored yet. However, as the system transits

Table 4.2 Main system parameters

Parameter	Numerical Value [Ref.]
C_i (IoT Sensors)	30 CPU-cycles/bit [230]
C_i (4-queens Puzzle)	87.8 CPU-cycles/bit [231]
C_i (5-queens Puzzle)	263 CPU-cycles/bit [231]
C_i (Face Recognition)	297.62 CPU-cycles/bit [232]
Channel width	80 MHz [233]
Communication band	5 GHz [233]
Communication mode	TCP [233]
D_i (IoT Sensors)	0.2 MB [230]
D_i (4-queens Puzzle)	0.2 MB [231]
D_i (5-queens Puzzle)	0.2 MB [231]
D_i (Face Recognition)	0.42 MB [232]
F_W	1 GHz [189], [234]
F_S	2.2 GHz [234]
Number of averaged simulations	10 per configuration
$P_{i,idleW}$	2.563 mW [235]
$P_{i,txW}$	255.2 mW [235]
$P_{i,rxS}$	210 mW [235]
Propagation loss model	Log distance [233]
Wi-Fi standard	IEEE 802.11ac [233]
α_c	10^{-28} [196]

through various states during the learning phase, the number of unexplored states reduces, and decision-making becomes increasingly refined. Within each state s, a particular action $a_{i,s}$ is selected, and the system advances to the next state, denoted as $s + 1$. During this transition, the immediate cost, denoted as $\psi(s, a_{i,s})$, is stored as feedback regarding the executed action. If this feedback is favorable, signifying that the task takes less time in execution and consumes a small amount of energy w.r.t the total sum of time and energy consumption in case of local execution, a higher Q-value is generated for that specific action. Conversely, the chosen action is penalized by decreasing its associated Q-value. This same reward/punishment policy applies to each state, and this iterative process repeats over a large number of iterations. At the end of learning phase, during which the system is expected to have explored all possible actions for all the states, the Q-table records the optimal action $a_{i,s}$ for each state s. Consequently, an optimal task offloading policy is determined.

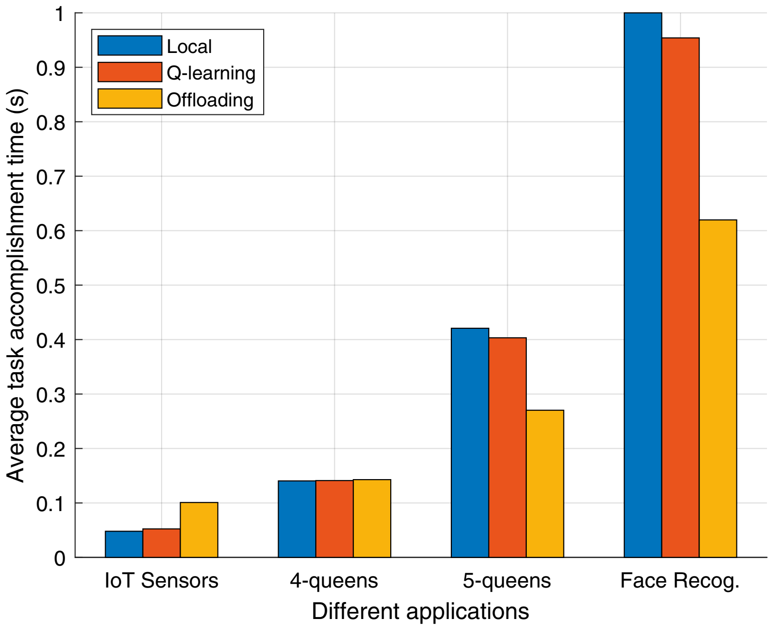

Figure 4.3 Average task accomplishment time for different applications for the three task execution scenarios ($\beta_E = \beta_T = 0.5$).

4.6 Performance Evaluation

In this section, we present the details regarding the simulation setup, considered performance parameters, and methods to evaluate our proposed task offloading solution. Subsequently, we present the comparative performance results of the proposed approach and discuss its effectiveness.

4.6.1 Simulation Environment

Our simulation experiments are performed in the ns-3 Network Simulator, that is a discrete-event network simulator built in C++ [233]. This open-source software offers realistic models for various networking protocols and standards while enjoying broad usage within the research community.

We deploy two nodes to represent a wearable device and a smartphone. We utilize the ns-3 Wi-Fi module to simulate the Wi-Fi connection between the wearable device and the smartphone, following the IEEE 802.11ac standard parameters with adhoc

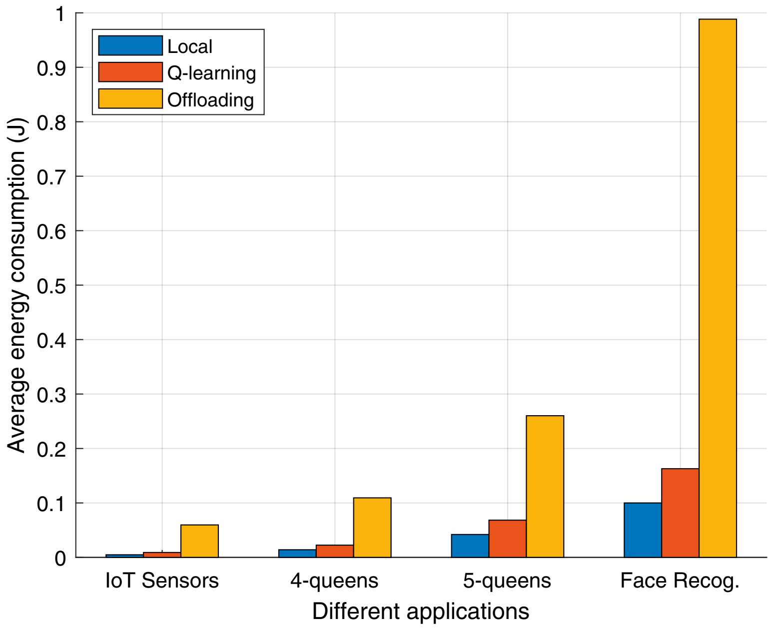

Figure 4.4 Average energy consumption for different applications for the three task execution scenarios ($\beta_E = \beta_T = 0.5$).

Wi-Fi MAC configuration. In addition, we maintain default settings for the physical layer, including a constant propagation delay and the log distance propagation loss models. Furthermore, we utilize the standard 5GHz band with an 80MHz channel width for the 802.11ac connection [236].

We present a comparison between our proposed Q-learning-based task offloading approach and two benchmark strategies, that are commonly used for performance evaluation [218]. These strategies, referred to as the *all-local* scheme and the *all-collaborative cloud* scheme in [218], have been renamed as the *Local* and *Offloading* schemes, respectively, to establish relevance with our proposed architecture. The operational details of these benchmark schemes are briefly outlined below:

- *Local:* In this strategy, all tasks that originate from the wearable device are executed locally without offloading.
- *Offloading:* In this approach, all tasks originating from the wearable device are offloaded and executed on the smartphone.

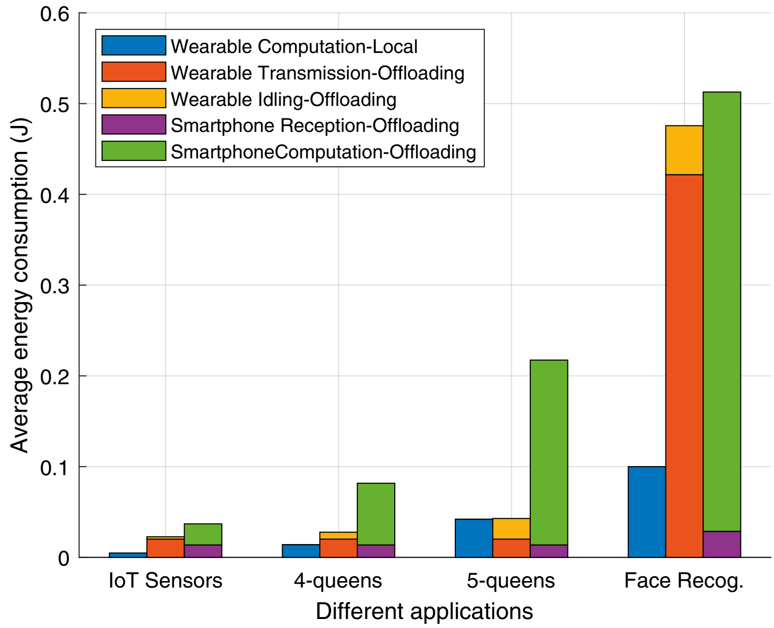

Figure 4.5 Average energy consumption breakdown for different applications for Local execution vs Offloading scenarios ($\beta_E = \beta_T = 0.5$).

We make the assumption that tasks originating from the wearable device can pertain to various applications, each characterized by distinct input data sizes and computational intensities. Consequently, we assess the performance of our proposed approach across multiple applications, specifically the IoT Sensors application, m-queens puzzle applications (with m values of 4 and 5), and a Face recognition application. The application m-queens is basically a generalization of the classic n-queens puzzle application, where 'm' stands for multiple. Table 4.2 provides details about the input data sizes and the corresponding computational intensities for each of these applications. In case of offloading, we use the Transmission Control Protocol (TCP) protocol for transferring task input data between devices, utilizing a bulk-send application within ns-3 to maximize the utilization of the transmission links. The main simulation parameters are summarized in table 4.2. Additionally, the computational overhead for Q-learning algorithms is relatively low [237]. Moreover, the energy and time overhead can vary significantly based on the hardware, software, and specific implementation details. Therefore, we have choosen to abstract away these factors to make the results more generalizable and applicable across different environments.

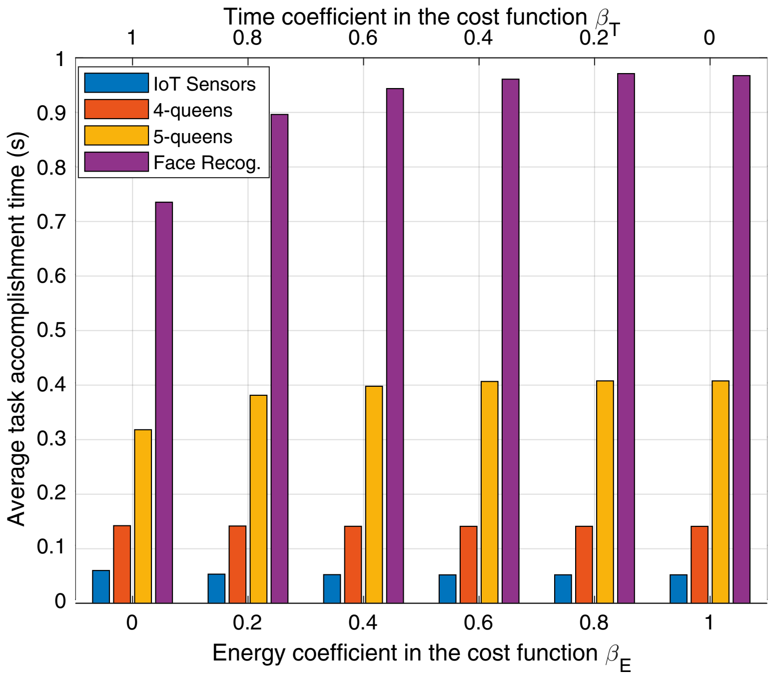

Figure 4.6 Average task accomplishment time per task using Q-learning for different applications when varying values of β_E and β_T.

4.6.2 Performance Metrics

The two main performance parameters include:

- *Task accomplishment time:* defined as the time since task generation at the wearable device to completion including both computation and communication time (in case of offloading).

- *Energy consumption:* modeled as the overall energy consumption of the system in execution of a task including both energy consumed at the wearable device as well as the smartphone (in case of offloading). It involves energy spent in task computation and communication including energy spent by the wearable during idling while the tasks get executed at the smartphone.

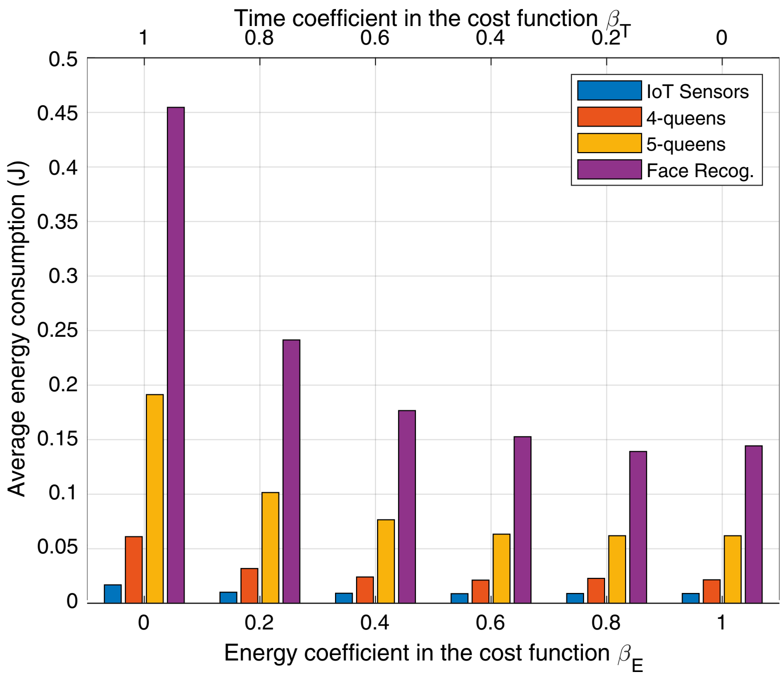

Figure 4.7 Average energy consumption per task using Q-learning for different applications when varying values of β_E and β_T.

4.6.3 Results and Discussion

Effect of Application Size on the overall performance

Figure 4.3 presents the average task accomplishment time per task across various applications under three different task execution scenarios: local execution, Q-learning, and offloading. To maintain a balance between time savings and energy conservation, we have kept the values of the time and energy coefficients, denoted as β_T and β_E, equal. The applications are arranged in ascending order based on their task input data size and computational intensity, with the IoT Sensors application being the lightest and the Face Recognition application being the heaviest among them.

For lighter applications, it is evident that task offloading does not yield a significant improvement in task accomplishment time compared to local execution. This is because the communication overhead, involved in transferring task data for execution on the smartphone, outweighs the computational gains. For instance, for the IoT Sensors application, Q-learning takes approximately 9% whereas, Offloading takes 110% more time as compared to the local execution. Consequently, for such lightweight tasks, the overall time savings are better achieved through local execution on the wearable device.

However, as we move to heavier applications, task offloading becomes a more viable option, as the limited computational capabilities of the wearable device substantially contribute to the overall task accomplishment time. Offloading these heavier tasks results in a considerable reduction in total time consumption. Notably, for the 4-queens application, all three schemes perform nearly identically. However, as we move beyond the 4-queens scenario and deal with heavier applications, offloading starts to offer significant time savings compared to local execution. For instance, for the Face Recognition application, Q-learning reduces the task accomplishment time by approximately 4.6% whereas Offloading reduces the task accomplishment time by 38%. Q-learning performs somewhere in between local execution and offloading since it runs some tasks locally as well in contrast to offloading where each task gets offloaded.

Nevertheless, opting for offloading every task turns out to be significantly more costly in terms of energy consumption because it involves energy consumed both in computation and communication. As a result, when examining figure 4.4, it becomes apparent that offloading consistently results in notably higher energy consumption compared to the other task execution scenarios for nearly all applications. For instance, for the IoT Sensors application, Q-learning offers approximately 84.85% more energy savings as compared to the Offloading approach. However, for the Face Recognition application Q-learning can achieve approximately 83.52% more energy savings as compared to Offloading.

Figure 4.5 presents a comparative breakdown of average energy consumption per task on both the wearable device and the smartphone for local versus offloading scenarios. In the case of local execution, the sole contributor to the overall energy consumption is the energy spent in computation of the task on the wearable device. Whereas, in the case of offloading, the total energy consumption on the wearable device includes energy consumed in transmitting input data to the smartphone and the energy consumed during idling while the task gets executed on the smartphone. In the case of offloading, the overall energy consumption on the smartphone includes energy spent in receiving the task from the wearable and executing it. For instance, for the Face Recognition application, it is observed that in the case of offloading, the

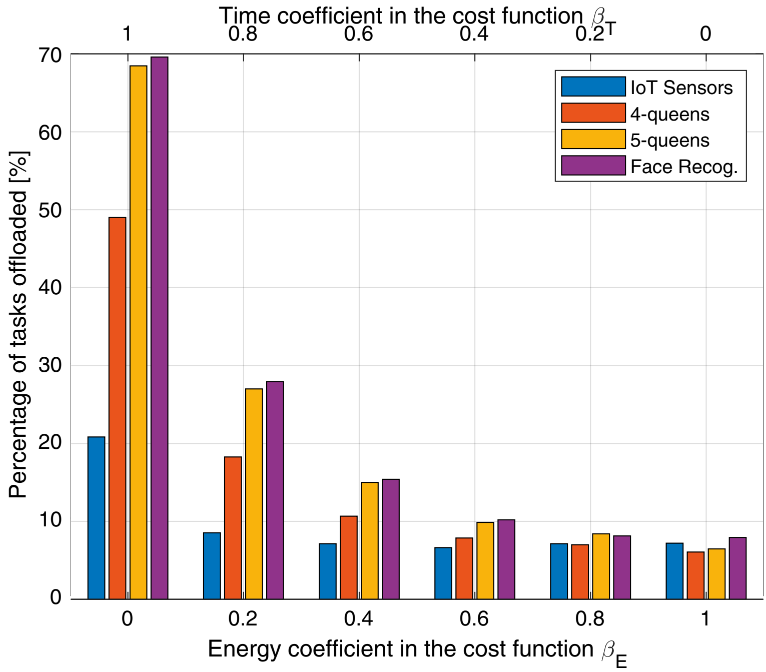

Figure 4.8 Percentage of tasks offloaded by the wearable device using Q-learning for different applications when varying values of β_E and β_T.

energy consumed by the wearable device during idling while the task gets executed on the smartphone is 11.35% of the overall energy consumption on the wearable device. Hence, it is worth noting that the energy consumed by the wearable device during idling while the task gets executed on the smartphone, is a non-negligible entity which gets even more pronounced as the computational intensity of the application increases.

Effect of Varying System Parameters

This subsection discusses the effect of varying system parameters namely β_E and β_T, the energy and time coefficients in the cost function given in equations 4.11 and 4.12.

Since there exists an inherent trade-off between task accomplishment time and energy consumption, it is important to fine-tune these parameters based on specific requirements. However, in this analysis, we aim to showcase the impact of varying these parameters on the overall system performance. In this set of graphs, we display both coefficients since their combined sum always equals one.

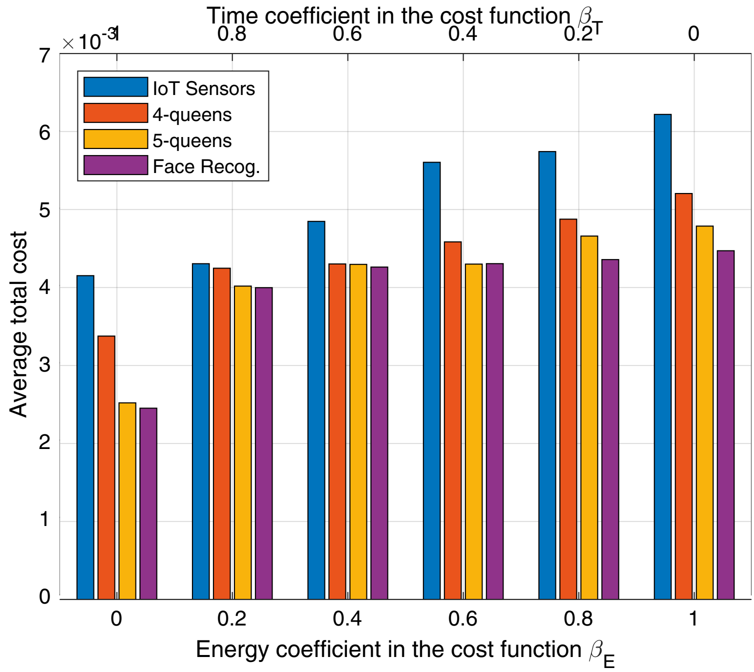

Figure 4.9 Average total cost per task using Q-learning for different applications when varying values of β_E and β_T.

Figure 4.6 illustrates the impact of varying the energy and time coefficients on the average task accomplishment time for each application using Q-learning. As the β_E value increases, it is noticeable that the average task accomplishment time also increases across all applications, reflecting a higher emphasis on energy conservation over time savings. This effect is particularly pronounced for heavier applications, primarily because heavier applications tend to offload a larger proportion of tasks. For instance, for the Face Recognition application, the average task accomplishment time increases by 31.55% if we compare the performance for $\beta_E = 0$ vs $\beta_E = 1$.

Similarly, in figure 4.7, we can observe the influence of changing the energy and time coefficients on the average energy consumption for each application using Q-learning. As the β_E value increases, the average energy consumption decreases for all applications, indicating a greater priority placed on energy conservation over time savings. This effect is more pronounced for heavier applications. For instance, comparing the performance for a Face Recognition application, the average energy consumption per task drops by approximately 68.26% moving from $\beta_E = 0$ to $\beta_E = 1$.

Figure 4.8 illustrates the percentage of tasks offloaded by the wearable device us-

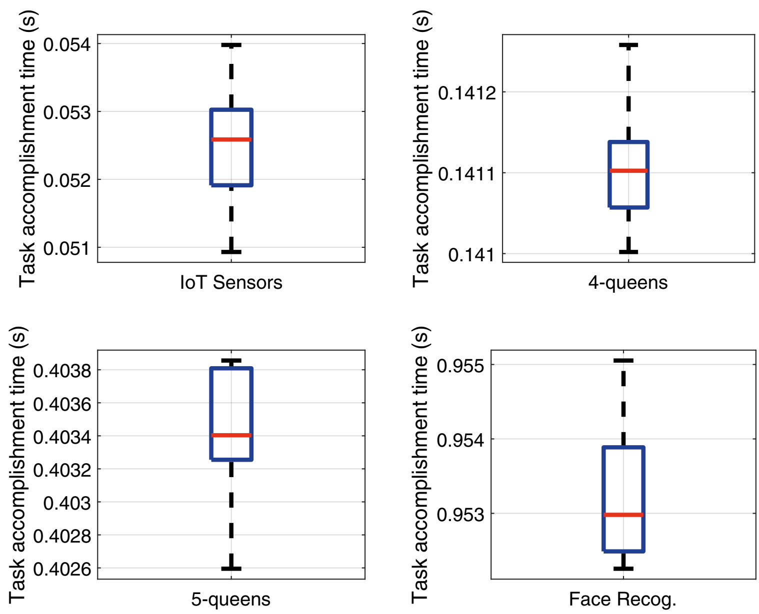

Figure 4.10 Task accomplishment time distribution for different applications using Q-learning (β_E = β_T = 0.5).

ing Q-learning across various applications while varying the values of β_E and β_T. It is evident that with a smaller β_E value (and consequently a higher β_T value), each application tends to offload a higher percentage of tasks. This behavior is particularly noticeable for heavier applications, as Q-learning strives to make near-optimal decisions. For lighter applications, where offloading does not yield substantial benefits in terms of both time and energy (as discussed in the previous subsection), the system tends to offload a lower percentage of tasks. However, as the size and complexity of the applications increase, the algorithm naturally tends to make more decisions in favor of offloading tasks, capitalizing on the abundant computational resources available at the edge. For instance, for $\beta_E = 0$, the wearable device offloads approximately 20% of tasks from the IoT Sensors application, in contrast to the Face Recognition application where the offloaded tasks reach around 70%. Conversely, when we increase the β_E value (and consequently reduce β_T), signifying a higher emphasis on energy conservation over time savings, each application attempts to offload a smaller percentage of tasks. Hence, it prioritizes local execution to conserve energy as observed from the graph for $\beta_E = 1$, the percentage of offloaded tasks remain under

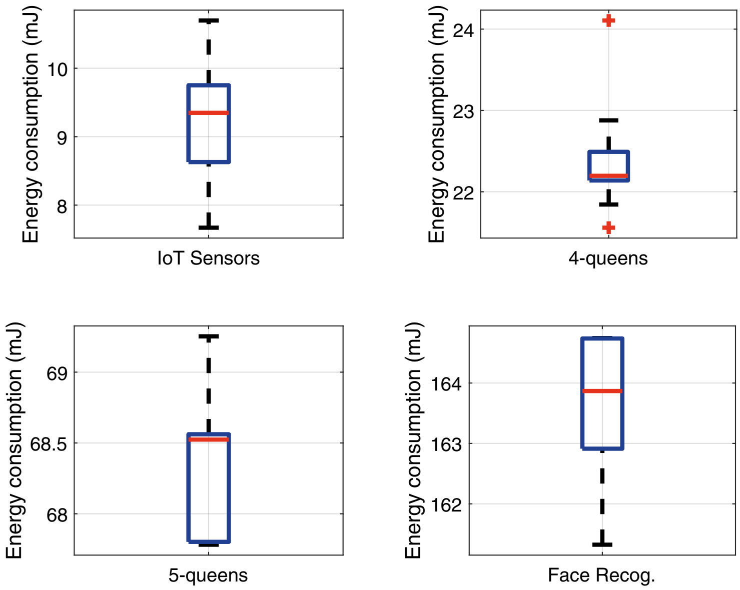

Figure 4.11 Energy consumption distribution for different applications using Q-learning ($\beta_E = \beta_T = 0.5$).

10% for all the applications.

Figure 4.9 depicts the influence of adjusting the energy and time coefficients on the average total cost, as defined in the cost function equations (4.11 and 4.12), for each application using Q-learning. When β_E is set to zero, it is evident that all applications tend to favor offloading a large number of tasks. However, this approach is not particularly beneficial for smaller applications. Consequently, the IoT Sensors application exhibits the highest cost compared to the other applications. For instance, for $\beta_E = 0$, the average total cost per task for the IoT Sensors applications is approximately 69.27% higher than the Face Recognition application. Moreover, as the β_E value increases, the overall cost rises for all applications, encompassing both computation and communication costs. Nevertheless, a consistent trend emerges, with offloading proving more advantageous when dealing with heavier applications that cannot be efficiently executed only using the computational resources available on the wearable device.

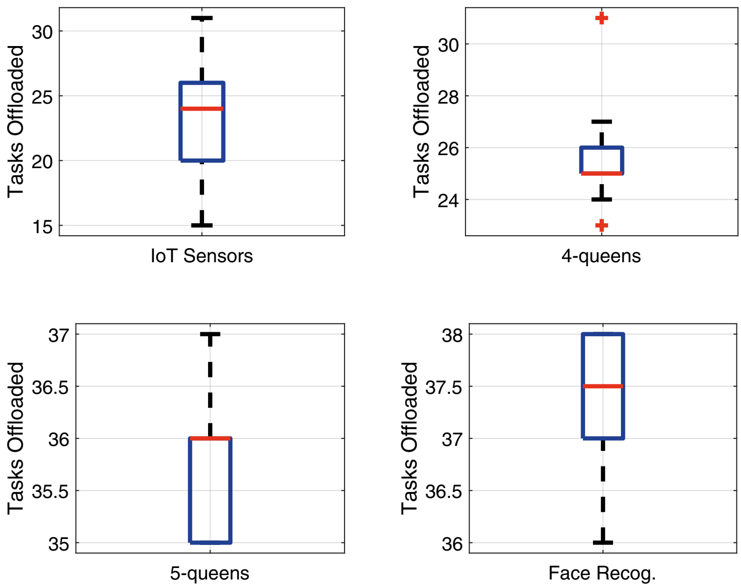

Figure 4.12 Task offloading distribution for different applications using Q-learning ($\beta_E = \beta_T = 0.5$).

Distribution graphs for different applications using Q-learning

Figures 4.10 and 4.11 depict the distribution of task accomplishment times and energy consumption across different applications when utilizing Q-learning with equal weightage for both time and energy savings. Among these applications, the IoT Sensors application, being the lightest in terms of computational demands, exhibits an average task accomplishment time of approximately 52.5ms, accompanied by an average energy consumption of around 9.5mJ. In contrast, the Face Recognition application, the most resource-intensive of all, demonstrates an average task accomplishment time of 953ms while consuming an average of roughly 164mJ of energy.

Figure 4.12 provides insight into the distribution of task offloading for various applications when employing Q-learning with equal weightage assigned to time and energy savings within the cost function. Notably, lighter applications such as the IoT Sensors application and the 4-queens application offload fewer tasks 24 and 25, respectively out of a total of 300 tasks generated at the wearable device to generate these statistics. However, as the computational demands of the application increase, even with equal weightage given to time and energy savings, the number of offloaded tasks rises.

4.7 Summary

This chapter introduces task offloading as an affective strategy for augmenting the capabilities of resource-limited IoWT devices, namely wearables, within edge computing networks. While various approaches address the TAP to optimize system resources, AI-based methods have emerged as the most promising solutions.

To summarize, we advocate the utilization of RL, a ML technique, to optimize the task offloading process. Within our proposed framework, a wearable device employs the Q-learning technique of RL to offload tasks to the paired smartphone, which acts as a deep edge node with comparatively greater computational and battery resources. In the Q-learning approach, the wearable device, also called the agent, learns iteratively by interacting with the environment to make intelligent task offloading decisions for each generated task.

We conducted a comprehensive analysis of various configurations to assess the impact of Q-learning on overall task accomplishment time and energy consumption across the involved devices. Our proposed framework was implemented in the ns-3 network simulator, enabling performance evaluation through simulation experiments across a range of applications. Additionally, we observed the effect of varying system parameters on the overall performance.

The results highlight a tradeoff between time savings and energy consumption. Notably, for lighter applications such as the IoT Sensors application, local computation outperforms both Q-learning and Offloading approaches in terms of task accomplishment time. This is due to the additional overhead introduced by the communication phase in both Q-learning and Offloading approaches. However, for a comparatively heavy application such as the Face Recognition application, Q-learning can bring approximately 26.45% savings in task accomplishment time instead of the local execution approach (setting $\beta_E = 0$, for maximum speedup). In comparison, approximately 85.4% energy savings are achievable as compared to the offloading approach (setting $\beta_E = 1$, for maximum energy savings) for the same application. Moreover, we demonstrate the impact of varying weight parameters for time and energy costs in the Q-learning algorithm on overall performance, including average task accomplishment time, average energy consumption, percentage of tasks offloaded, and total cost. These results offer valuable insights into fine-tuning of parameters for optimal performance. Finally, distribution graphs illustrating performance parameters such as task accomplishment time, energy consumption, and tasks offloaded are presented for different applications. Analyzing these distribution graphs, it is worth noting that leveraging Q-learning to offload tasks to a nearby edge device, such as a smartphone, allows achieving a task accomplishment time of 0.953s for a heavier application like Face Recognition at the expense of merely 164mJ of energy.

5 CONCLUSION

This chapter concludes the thesis by providing a summary of the main research findings. Subsequently, we highlight potential avenues for future research along with the major challenges associated.

5.1 Main Research Findings

Wearable technology is rapidly gaining popularity among consumers, finding utility in a diverse range of value-added and entertainment applications. These applications include health monitoring, fitness tracking, human activity recognition, VR, AR, gaming, and more. As the adoption of wearables continues to grow, significant technological advancements have been made to enhance the overall efficiency of these devices, ultimately improving the user experience. However, despite these advancements, the wearable industry faces several challenges that restrict its continued evolution.

In this thesis, we highlighted that many of those challenges stem from two primary limitations: i) the constrained battery life of wearable devices, and ii) the insufficient computational capacity to accommodate the requirements of increasingly sophisticated modern applications. In this context, we formulated a set of research questions (reproduced below) to address in this thesis.

Q1. *What is the current state of research focused on energy efficiency in the IoWT technology, including year-wise publication trends, main application areas, performance parameters, evaluation tools, prevalent wireless communication technologies, and strategies for enhancing energy efficiency?*

To address research question Q1, we conducted a comprehensive SLR of state-of-the-art solutions focused on enhancing energy efficiency within the IoWT domain, as detailed in chapter 2. We proposed a taxonomy for categorizing IoWT solutions based on their targeted application areas from an energy efficiency perspective, classifying them into healthcare, activity recognition, smart environments, and general solutions. Given the historical development of wearables for specific medical purposes, the review revealed that a substantial portion of the existing solutions focused on healthcare applications. However, recent advancements have expanded the use of wearables into a multitude of other domains beyond healthcare. Additionally, we presented a statistical analysis of these solutions over the years, analyzing the publication trends. This analysis revealed a continuous growth in research related to wearables, signaling an increasing interest in this field. Moreover, a thorough qualitative and comparative analysis of existing studies within each category is provided highlighting the merits, demerits, main performance parameters, and major contributions of each solution. Additionally, we performed a statistical analysis to identify the prevalent tools used for assessing the performance of proposed solutions. This analysis revealed a general trend among researchers to develop prototypes for validating the effectiveness of their proposed solutions. However, some studies presented simulations-based results, with MATLAB emerging as the most frequently used simulator. Similarly, we presented another statistical analysis to highlight the most commonly used communication technologies in wearables. This analysis demonstrated that BLE was the dominant choice, primarily due to its low power consumption characteristics. Finally, a detailed discussion is provided outlining the main strategies found in the literature for enhancing energy efficiency in wearables while also emphasizing the challenges involved.

Q2. *What are the potential benefits and limitations of task offloading for wearables in multi-tier edge architectures in terms of task accomplishment time and energy consumption, considering realistic settings regarding computing task requirements, device capabilities, and inter-device distance?*

To address research question Q2, chapter 3 investigates the benefits and limitations of task offloading for wearables. A two-tier edge architecture is proposed that comprises a smartphone and an edge server as task executor nodes. Based on the proposed mathematical formulation, a detailed numerical analysis is presented focusing on the task accomplishment time and energy consumption parameters. We evaluated the performance for a wide range of different settings. These settings include varying input data sizes as well as the required computational intensities for tasks to observe the effect on overall performance. Additionally, we also analyzed the performance by varying device capabilities in terms of CPU computational capacity. Moreover, considering the mobility of a user, we analyzed the overall performance by varying spatial separation between the smartphone and edge server. Furthermore, since task offloading involves both computation and communication phases at the devices involved, therefore, we provide a breakdown of overall time and energy consumption for different task execution scenarios to illustrate the individual contribution of each phase.

Based on the considered settings, we observed that offloading tasks to the edge server generally proved to be better as compared to both local execution and offloading to the smartphone. However, when the smartphone is situated at the cell border, experiencing challenging signal propagation conditions, executing tasks on the smartphone becomes advantageous over offloading to the edge unless the smartphone's battery is running low or the task is exceptionally computationally intensive. Finally, opting for local task execution on the wearable is beneficial when dealing with tasks that are not computationally intensive. This is due to the fact that the delay incurred during the transfer of input data over wireless links can potentially surpass the total time required to execute the task.

Q3. *How can Q-learning, a RL-based technique, be effectively utilized to optimize task offloading for wearables in an edge computing framework to enhance their overall performance, network resource utilization, and end-user experience?*

To address research question Q3, chapter 4 presents a background on several task offloading strategies available in the literature aimed at addressing the TAP to optimize system resources. Among the available strategies, AI-based techniques were found to be the most promising solutions. Specifically, the model-free Q-learning technique of RL, is found to be the most suitable and affective for wearables due to its inherent simplicity. We propose an edge computing framework where the wearable device utilizes the model-free Q-learning approach of RL to make intelligent task offloading decisions leveraging the resources available on the smartphone that serves as a deep edge server. In this approach, the wearable device (a.k.a. the agent), learns iteratively by interacting with the environment to make intelligent task offloading decisions for each generated task without prior knowledge. The effectiveness of the proposed framework is evaluated through extensive simulation experiments conducted in the ns-3 network simulator. A comprehensive analysis of various configurations to assess the impact of Q-learning on overall task accomplishment time and energy consumption is presented across the involved devices. We consider several different applications namely the IoT sensors application, m-queens puzzle applications (with m=4 and m=5), and a face recognition application, with different task input data sizes and task computational intensities while also highlighting how varying the main system parameters of the Q-learning algorithm affects overall performance.

The results demonstrate a tradeoff between time savings and energy consumption. Notably, for more demanding applications such as Face Recognition, Q-learning can bring approximately 26.45% savings in task accomplishment time compared to local execution (with β_E set to 0, for maximum speedup). Moreover, approximately 85.4% energy savings can be achieved compared to the offloading approach (with β_E set to 1, for maximum energy savings) for the same application. We also illustrate the impact of varying weight parameters for time and energy costs in the Q-learning algorithm on overall performance, including average task accomplishment time, average energy consumption, percentage of tasks offloaded, and total cost. These results provide valuable insights into the fine-tuning of parameters for optimal performance. Lastly, distribution graphs depicting performance parameters such as task accomplishment time, energy consumption, and tasks offloaded are presented for different applications. It is worth noting that leveraging Q-learning to offload tasks to a nearby edge device, such as a smartphone, enables achieving a task accomplishment time of 0.953s for a heavier application like Face Recognition at the expense of merely 164mJ of energy.

To summarize, we performed an extensive SLR focussed on the energy efficiency aspect of wearables. We identified two major challenges associated with wearable development, namely the limited battery power and insufficient computation power available on the contemporary wearable devices available in the market. We gathered diverse statistical data from the literature, providing insights into the research aimed at improving the overall energy efficiency of the wearables. We also highlighted several strategies available in the literature to improve the energy efficiency of wearables. Subsequently, we focussed on the task offloading strategy to address the aforementioned main challenges. We performed a detailed numerical analysis to quantify the benefits task offloading can bring to improve the overall performance of the wearables while also highlighting the limitations. Moreover, we explored the potential application of ML techniques to automate the task offloading process on wearable devices. Furthermore, we presented an edge computing framework utilizing the model free Q-learning technique of RL to improve the task offloading process and assessed its performance through extensive simulations.

5.2 Challenges and Future Research Directions

While it is true that numerous solutions addressing various aspects within the IoWT domain already exist, we foresee that this field has yet to reach its full potential. We anticipate that opportunities for further enhancement and research still exist. Therefore, we have highlighted avenues for further research within this domain, with the aim of inspiring researchers to develop more capable IoWT-based systems to meet the upcoming user demands.

5.2.1 Parallel Task Execution and Split Computing

To further enhance the benefits of task offloading in the context of IoWT, exploring parallel task execution for multitasking support, can be another meaningful research direction. In order to efficiently manage multiple applications, it becomes crucial for the wearable device to promptly decide the execution location of each generated task. These decisions should consider both the specific application requirements and the availability of resources to optimize overall performance. Furthermore, another closely related potential strategy involves breaking down individual tasks into subtasks, each of which can be executed at distinct network entities. Such split computing strategies to execute tasks partially at the edge server and the smartphone/wearable can certainly lower overall task execution time while also optimizing the energy consumption. However, managing inter-task dependencies and resource utilization requires careful consideration in the implementation. Additionally, the division of tasks into subtasks should ensure that each atomic subtask can function independently, thereby enhancing the overall user experience.

5.2.2 Approximate Computing

With the advancements in processor design, it is anticipated that future wearable devices will be equipped with more powerful processors with substantial storage resources. However, the energy consumption also increases significantly with the increase in computation capacity. Hence, it is crucial to manage these resources efficiently to conserve energy. Moreover, the growing demand for high computing resources across various wearable applications necessitates the development of efficient computing techniques. Therefore, utilizing approximate computing in conjunction with task offloading is an innovative approach that can significantly impact the efficiency and performance of computational tasks, particularly in resource-constrained environments like those found in wearable devices within the context IoWT. These approaches involve trading output accuracy for gains in computing time and energy by relying on nearly accurate results. However, it necessitates a thoughtful analysis of task characteristics and precision requirements to achieve the right balance between computational accuracy and performance gains (for a detailed discussion on approximate computing and the tradeoffs, refer to section 2.3.9).

5.2.3 Direct Internet Connectivity

From the communications perspective, a significant number of current wearable devices establish their Internet connection via a gateway node, often a smartphone, due to the absence of direct long-range connectivity capabilities. However, such an arrangement can lead to considerable performance limitations, particularly for high-end wearables that demand high data rates. Examples include applications in fields like AR/VR/MR or XR applications. Consequently, there is a growing expectation that devices with direct Internet connectivity, equipped with IEEE 802.11 and/or cellular modules, will receive more attention in the immediate future. Moreover, the advent of long-range non-cellular connectivity solutions such as NB-IoT, LoRa, Sigfox, among others, is anticipated to enter the wearable industry, thereby paving the way for a multitude of novel IoWT applications. However, direct Internet connectivity has the potential to introduce new challenges regarding the energy efficiency of wearable devices and monetary costs. Direct communication with an access point (as in Wi-Fi) or a base station (in the case of cellular/non-cellular connectivity) would involve transmitting data over comparatively longer distances with high transmission powers. Moreover, offloading tasks using cellular/non-cellular connectivity options can also increase the data costs. Therefore, another interesting future research direction can be to develop user-centric RL-based task offloading algorithms that take into account user preferences and goals when making offloading decisions along with adaptive transmission power control mechanisms.

5.2.4 Personalized Wearable Clouds

Generally, the majority of available wearables currently operate as standalone devices, performing their designated tasks individually without any substantial collaboration with other wearables. Nevertheless, as consumer interest in integrating wearables into their everyday activities grows, it is anticipated that individuals will incorporate multiple wearables into their daily routines in the near future. Consequently, a network of wearables could be established, wherein personal wearables could capitalize on the sensing, computing, and transmission capabilities of nearby wearables. This collaborative approach could enable the efficient execution of desired tasks by forming personalized wearable clouds through collaboration. Hence necessitating the need for developing multi-agent distributed Q-learning techniques that can efficiently operate in environments with multiple interconnected devices.

5.2.5 Security and Privacy

It is crucial to highlight that security and privacy have emerged as paramount concerns, particularly within the context of medical applications. Wearable devices frequently carry sensitive and private user data, which could potentially be exploited to identify and trace individuals. For instance, various wearables implement specific biometric-based locking mechanisms, such as fingerprinting and facial recognition techniques. Notably, the biometric data associated with users constitutes the most sensitive information, as compared to passwords which can be changed; in contrast, biometric or behavioral traits often remain constant throughout an individual's lifetime. It has been observed that many commercially available wearables lack comprehensive security features, often due to concerns about performance degradation. This is because many data encryption and security methods available are computationally demanding for wearable devices. Consequently, the development of security and privacy techniques that are lightweight and efficient, tailored specifically for wearables, has emerged as a significant research domain. Moreover, within the context of task offloading for wearables, developing RL-based algorithms that consider security risks and privacy concerns when making offloading decisions can be a potential research direction. This becomes particularly important in scenarios involving sensitive data such as healthcare and services dealing with confidential information.

5.2.6 Energy Harvesting

With the evolution of processing units and the rise of high-performance wearables reliant on intensive computations, are contributing to the concern that current battery power resources might prove inadequate for prolonged device operation. Consequently, it is anticipated that EH will become a crucial component of future high-powered wearables. Many researchers are actively exploring ideas to enable this feature. These include microkinetic EH systems that leverage frequencies found in human motion to extract energy, wearable power through solar EH, the development of self-powering smart fabrics, and even wireless power transfer for implantable devices. However, integrating EH techniques into the IoWT still faces several challenges. Firstly, the EH efficiencies of current state-of-the-art harvesters are insufficient to independently power wearable devices in the IoWT. Secondly, ambient energy availability is not consistent. Thirdly, the compact design goals of wearable devices pose an additional hurdle, as EH necessitates the integration of various hardware components like ambient energy harvesters and supplementary batteries. Consequently, extensive research is necessary to enable future IoWT devices to reliably generate power from ambient sources. Hence, extending the battery life of wearables.

5.2.7 Proximity Detection

In the context of wearables, proximity detection refers to the capabilities of wearable devices to sense and measure the distance or proximity to other objects/devices in their vicinity. Proximity detection is a promising area of research with the potential to revolutionize various fields, such as social interaction and health monitoring, where wearable devices equipped with proximity sensors can help individuals maintain safe distances from others, especially in situations like crowded public spaces or during contagious disease outbreaks. Similarly, they can significantly benefit individuals with disabilities, such as the visually impaired. Additionally, wearables can aid in environmental research by measuring the proximity to various environmental factors like pollution sources or radiation. Moreover, they can also be used to enhance workplace safety as well as to locate and place workers efficiently. Furthermore, proximity detection can transform retail and marketing strategies by enabling businesses to engage with customers based on their proximity to products or store areas, offering targeted promotions or product information. Hence, the potential applications of proximity detection using wearable devices are vast, and ongoing research and development in this area have the potential to bring about transformative changes in how we interact with our environment, technology, and each other. However, gadgets of this nature need to possess sufficient resources to efficiently carry out the desired task as well as to effectively support the user throughout the entire day. Therefore, the utilization of low-power technologies in the design of upcoming proximity detection wearables becomes a critical consideration and an active research area.

5.2.8 Comfort and ease-of-use

The comfort and user-friendliness of wearables are of paramount importance. Considering the close proximity of these devices to the human body and skin, their design must carefully account for these factors. Specifically, the risk of overheating or short circuits should be carefully addressed when designing future wearables. Some high-end wearables presently available generate substantial heat, potentially preventing widespread consumer acceptance. Hence, these factors stand as crucial elements to be considered during the development of upcoming wearable technologies.

REFERENCES

[1] W. B. Qaim, A. Ometov, A. Molinaro, *et al.*, "Towards Energy Efficiency in the Internet of Wearable Things: A Systematic Review," *IEEE Access*, 2020.

[2] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Internet of Things Applications: A Systematic Review," *Computer Networks*, 2019.

[3] F. J. Dian, R. Vahidnia, and A. Rahmati, "Wearables and the Internet of Things (IoT), Applications, Opportunities, and Challenges: A Survey," *IEEE Access*, 2020.

[4] S. Hiremath, G. Yang, and K. Mankodiya, "Wearable Internet of Things: Concept, Architectural Components and Promises for Person-Centered Healthcare," *IEEE MOBIHEALTH*, 2014.

[5] M. Cornacchia, K. Ozcan, Y. Zheng, and S. Velipasalar, "A Survey on Activity Detection and Classification Using Wearable Sensors," *IEEE Sensors Journal*, 2016.

[6] G. Fortino, D. Parisi, V. Pirrone, and G. Di Fatta, "BodyCloud: A SaaS Approach for Community Body Sensor Networks," *Future Generation Computer Systems*, 2014.

[7] B. Latre, B. Braem, I. Moerman, C. Blondia, and P. Demeester, "A survey on wireless body area networks," *Wireless networks*, 2011.

[8] S. Seneviratne, Y. Hu, T. Nguyen, *et al.*, "A Survey of Wearable Devices and Challenges," *IEEE Communications Surveys & Tutorials*, 2017.

[9] A. Ometov, D. Kozyrev, V. Rykov, S. Andreev, Y. Gaidamaka, and Y. Koucheryavy, "Reliability-Centric Analysis of Offloaded Computation in Cooperative Wearable Applications," *Wireless Communications and Mobile Computing*, 2017.

[10] Y. Wang, B. Millet, and J. L. Smith, "Designing Wearable Vibrotactile Notifications for Information Communication," *International Journal of Human-Computer Studies*, 2016.

[11] L. Piwek, D. A. Ellis, S. Andrews, and A. Joinson, "The Rise of Consumer Health Wearables: Promises and Barriers," *PLoS Medicine*, 2016.

[12] J. Lee, D. Kim, H.-Y. Ryoo, and B.-S. Shin, "Sustainable Wearables: Wearable Technology for Enhancing the Quality of Human Life," *Sustainability*, 2016.

[13] Business Wire, Inc., *The Wearable Technology Ecosystem: 2016-2030 - Opportunities, Challenges, Strategies, Industry Verticals and Forecasts for the $40 Billion Market - Research and Markets*, [Online] (accessed 2024-03-13) https://www.businesswire.com/news/home/20160504005813/en/Wearable-Technology-Ecosystem-2016-2030---Opportunities-Challenges, May 2016.

[14] I-SCOOP, *Wearables Market Outlook 2020: Drivers and New Markets*, [Online] (accessed 2024-03-13) https://www.i-scoop.eu/wearables-market-outlook-2020-drivers-new-markets/, 2016.

[15] J. Hayward, *Wearable Sensors 2018-2028: Technologies, Markets & Players*. IDTechEx Ltd., 2018.

[16] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, and D. S. Nikolopoulos, "Challenges and Opportunities in Edge Computing," *IEEE SmartCloud*, 2016.

[17] A. Pyattaev, K. Johnsson, S. Andreev, and Y. Koucheryavy, "Communication Challenges in High-Density Deployments of Wearable Wireless Devices," *IEEE Wireless Communications*, 2015.

[18] R. Rawassizadeh, B. A. Price, and M. Petre, "Wearables: Has the Age of Smartwatches Finally Arrived?" *ACM Communications*, 2014.

[19] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet of Things Journal*, 2017.

[20] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys & Tutorials*, 2017.

[21] M. Chen and Y. Hao, "Task Offloading for Mobile Edge Computing in Software Defined Ultra-Dense Network," *IEEE Journal on Selected Areas in Communications*, 2018.

[22] M. R. Nakhkash, T. N. Gia, I. Azimi, A. Anzanpour, A. M. Rahmani, and P. Liljeberg, "Analysis of Performance and Energy Consumption of Wearable Devices and Mobile Gateways in IoT Applications," *International Conference on Omni-Layer Intelligent Systems*, 2019.

[23] H. Hua, Y. Li, T. Wang, N. Dong, W. Li, and J. Cao, "Edge computing with artificial intelligence: A machine learning perspective," *ACM Computing Surveys*, 2023.

[24] D. Hortelano, I. de Miguel, R. J. D. Barroso, *et al.*, "A comprehensive survey on reinforcement-learning-based computation offloading techniques in edge computing systems," *Journal of Network and Computer Applications*, 2023.

[25] J. Williamson, Q. Liu, F. Lu, *et al.*, "Data Sensing and Analysis: Challenges for Wearables," *IEEE Asia and South Pacific Design Automation Conference*, 2015.

[26] C. Min, S. Kang, C. Yoo, *et al.*, "Exploring current practices for battery use and management of smartwatches," *ACM International Symposium on Wearable Computers*, 2015.

[27] T. Rault, A. Bouabdallah, Y. Challal, and F. Marin, "A Survey of Energy-Efficient Context Recognition Systems Using Wearable Sensors for Healthcare Applications," *Pervasive and Mobile Computing*, 2017.

[28] H. Sun, Z. Zhang, R. Q. Hu, and Y. Qian, "Wearable Communications in 5G: Challenges and Enabling Technologies," *IEEE Vehicular Technology Magazine*, 2018.

[29] A. Pantelopoulos and N. G. Bourbakis, "A Survey on Wearable Sensor-based Systems for Health Monitoring and Prognosis," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 2009.

[30] E. Sazonov, *Wearable Sensors: Fundamentals, Implementation and Applications*. Elsevier, 2014.

[31] O. D. Lara and M. A. Labrador, "A Survey on Human Activity Recognition Using Wearable Sensors," *IEEE Communications Surveys & Tutorials*, 2012.

[32] D. Dakopoulos and N. G. Bourbakis, "Wearable Obstacle Avoidance Electronic Travel Aids for Blind: A Survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 2009.

[33] X. Zhang, Z. Yang, W. Sun, *et al.*, "Incentives for Mobile Crowd Sensing: A Survey," *IEEE Communications Surveys & Tutorials*, 2015.

[34] K. Hartman, *Make: Wearable Electronics: Design, Prototype, and Wear Your Own Interactive Garments*. Maker Media, Inc., 2014.

[35] R. Harle, "A Survey of Indoor Inertial Positioning Systems for Pedestrians," *IEEE Communications Surveys & Tutorials*, 2013.

[36] A. Liberati, D. G. Altman, J. Tetzlaff, *et al.*, "The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate health care interventions: explanation and elaboration," *Journal of clinical epidemiology*, 2009.

[37] *Scopus*, [Online] Available: https://www.scopus.com/home.uri (Accessed 2024-03-13).

[38] *Web of Science*, [Online] Available: https://www.webofscience.com/wos/woscc/basic-search (Accessed 2024-03-13).

[39] A. K. Yetisen, J. L. Martinez-Hurtado, B. Ünal, A. Khademhosseini, and H. Butt, "Wearables in Medicine," *Advanced Materials*, 2018.

[40] E. Mencarini, A. Rapp, L. Tirabeni, and M. Zancanaro, "Designing Wearable Systems for Sports: A Review of Trends and Opportunities in Human–Computer Interaction," *IEEE Transactions on Human-Machine Systems*, 2019.

[41] Y. Jiang, Z. Li, and J. Wang, "Ptrack: Enhancing the Applicability of Pedestrian Tracking with Wearables," *IEEE Transactions on Mobile Computing*, 2018.

[42] L. Sharma, A. Javali, R. Nyamangoudar, R. Priya, P. Mishra, and S. K. Routray, "An Update on Location Based Services: Current State and Future Prospects," *IEEE ICCMC*, 2017.

[43] N. Siddiqui and R. H. Chan, "A Wearable Hand Gesture Recognition Device Based on Acoustic Measurements at Wrist," *IEEE EMBC*, 2017.

[44] M. Evers, M. Krzywdzinski, and S. Pfeiffer, "Designing Wearables for Use in the Workplace: The Role of Solution Developers," WZB Discussion Paper SP III 2018-301, Germany, Tech. Rep., 2018.

[45] K. Akkaya, I. Guvenc, R. Aygun, N. Pala, and A. Kadri, "IoT-based Occupancy Monitoring Techniques for Energy-Efficient Smart Buildings," *IEEE WCNCW*, 2015.

[46] A. De Paola, M. Ortolani, G. Lo Re, G. Anastasi, and S. K. Das, "Intelligent Management Systems for Energy Efficiency in Buildings: A Survey," *ACM Computing Surveys*, 2014.

[47] T. A. Nguyen and M. Aiello, "Energy Intelligent Buildings Based on User Activity: A Survey," *Energy and Buildings*, 2013.

[48] J. S. de Oliveira Neto, A. L. M. Silva, F. Nakano, J. J. Pérez-Álcazar, and S. T. Kofuji, "When Wearable Computing Meets Smart Cities: Assistive Technology Empowering Persons with Disabilities," *Smart Cities and Smart Spaces: Concepts, Methodologies, Tools, and Applications*, 2019.

[49] H. Djelouat, M. Al Disi, I. Boukhenoufa, *et al.*, "Real-time ECG Monitoring Using Compressive Sensing on a Heterogeneous Multicore Edge-Device," *Microprocessors and Microsystems*, 2020.

[50] C. Pereira, J. Mesquita, D. Guimarães, F. Santos, L. Almeida, and A. Aguiar, "Open IoT Architecture for Continuous Patient Monitoring in Emergency Wards," *Electronics*, 2019.

[51] K. M. Awan, N. Ashraf, M. Q. Saleem, *et al.*, "A Priority-based Congestion-Avoidance Routing Protocol Using IoT-based Heterogeneous Medical Sensors for Energy Efficiency in Healthcare Wireless Body Area Networks," *International Journal of Distributed Sensor Networks*, 2019.

[52] C. Wang, Y. Qin, H. Jin, *et al.*, "A Low Power Cardiovascular Healthcare System with Cross-Layer Optimization from Sensing Patch to Cloud Platform," *IEEE Transactions on Biomedical Circuits and Systems*, 2019.

[53] M. S. Mahmud, H. Fang, and H. Wang, "An Integrated Wearable Sensor for Unobtrusive Continuous Measurement of Autonomic Nervous System," *IEEE Internet of Things Journal*, 2018.

[54] M. Roy, C. Chowdhury, and N. Aslam, “Designing Transmission Strategies for Enhancing Communications in Medical IoT Using Markov Decision Process,” *Sensors*, 2018.

[55] M. Anwar, A. H. Abdullah, A. Altameem, *et al.*, “Green Communication for Wireless Body Area Networks: Energy Aware Link Efficient Routing Approach,” *Sensors*, 2018.

[56] C. Beach, S. Krachunov, J. Pope, *et al.*, “An Ultra Low Power Personalizable Wrist Worn ECG Monitor Integrated with IoT Infrastructure,” *IEEE Access*, 2018.

[57] M. S. Mahmud, H. Wang, and H. Fang, “SensoRing: An Integrated Wearable System for Continuous Measurement of Physiological Biomarkers,” *IEEE ICC*, 2018.

[58] G. Yang, J. Deng, G. Pang, *et al.*, “An IoT-enabled Stroke Rehabilitation System Based on Smart Wearable Armband and Machine Learning,” *IEEE Journal of Translational Engineering in Health and Medicine*, 2018.

[59] M. Al Disi, H. Djelouat, C. Kotroni, *et al.*, “ECG Signal Reconstruction on the IoT-Gateway and Efficacy of Compressive Sensing Under Real-Time Constraints,” *IEEE Access*, 2018.

[60] M. Hooshmand, D. Zordan, D. Del Testa, E. Grisan, and M. Rossi, “Boosting the Battery Life of Wearables for Health Monitoring through the Compression of Biosignals,” *IEEE Internet of Things Journal*, 2017.

[61] S. Preejith, A. Jeelani, P. Maniyar, J. Joseph, and M. Sivaprakasam, “Accelerometer based System for Continuous Respiratory Rate Monitoring,” *IEEE MeMeA*, 2017.

[62] K. Xu, Y. Li, and F. Ren, “A Data-driven Compressive Sensing Framework Tailored for Energy-Efficient Wearable Sensing,” *IEEE ICASSP*, 2017.

[63] H. Djelouat, H. Baali, A. Amira, and F. Bensaali, “An Adaptive Joint Sparsity Recovery for Compressive Sensing Based EEG System,” *Wireless Communications and Mobile Computing*, 2017.

[64] F. Touati, A. B. Mnaouer, O. Erdene-Ochir, W. Mehmood, A. Hassan, and B. Gaabab, "Feasibility and Performance Evaluation of a 6LoWPAN-enabled Platform for Ubiquitous Healthcare Monitoring," *Wireless Communications and Mobile Computing*, 2016.

[65] L. Hu, Y. Zhang, D. Feng, M. M. Hassan, A. Alelaiwi, and A. Alamri, "Design of QoS-aware Multi-level MAC-layer for Wireless Body Area Network," *Journal of Medical Systems*, 2015.

[66] T.-Y. Kim and E.-J. Kim, "Multi-Hop WBAN Configuration Approach for Wearable Machine-to-Machine Systems," *Multimedia Tools and Applications*, 2016.

[67] V. K. Sarker, M. Jiang, T. N. Gia, A. Anzanpour, A. M. Rahmani, and P. Liljeberg, "Portable Multipurpose Bio-Signal Acquisition and Wireless Streaming Device for Wearables," *IEEE SAS*, 2017.

[68] R. Gao, M. Zhao, Z. Qiu, Y. Yu, and C. H. Chang, "Web-based Motion Detection System for Health Care," *IEEE ICIS*, 2015.

[69] Y. Liao, Q. Yu, Y. Han, and M. S. Leeson, "Relay-Enabled Task Offloading Management for Wireless Body Area Networks," *Applied Sciences*, 2018.

[70] V. P. Rachim and W.-Y. Chung, "Wearable nonContact Armband for Mobile ECG Monitoring System," *IEEE Transactions on Biomedical Circuits and Systems*, 2016.

[71] E. Aguirre, P. Lopez-Iturri, L. Azpilicueta, *et al.*, "Design and Performance Analysis of Wireless Body Area Networks in Complex Indoor e-Health Hospital Environments for Patient Remote Monitoring," *International Journal of Distributed Sensor Networks*, 2016.

[72] H. Dagale, S. Anand, M. Hegde, *et al.*, "CyPhyS+: A Reliable and Managed Cyber-Physical System for Old-Age Home Healthcare over a 6LoWPAN Using Wearable Motes," *IEEE International Conference on Services Computing*, 2015.

[73] S. B. Baker, W. Xiang, and I. Atkinson, "Internet of Things for Smart Healthcare: Technologies, Challenges, and Opportunities," *IEEE Access*, 2017.

[74] Y. Liu, M. Pharr, and G. A. Salvatore, "Lab-on-Skin: A Review of Flexible and Stretchable Electronics for Wearable Health Monitoring," *ACS Nano*, 2017.

[75] S. Pinto, J. Cabral, and T. Gomes, "We-care: An IoT-based Health Care System for Elderly People," *IEEE ICIT*, 2017.

[76] H. Djelouat, H. Baali, A. Amira, and F. Bensaali, "IoT Based Compressive Sensing for ECG Monitoring," *IEEE iThings, GreenCom, CPSCom, and SmartData*, 2017.

[77] A. Jordao, A. C. Nazare Jr, J. Sena, and W. R. Schwartz, "Human Activity Recognition Based on Wearable Sensor Data: A Standardization of the State-of-the-Art," *arXiv preprint arXiv:1806.05226*, 2018.

[78] N. Al-Naffakh, N. Clarke, P. Dowland, and F. Li, "Activity Recognition Using Wearable Computing," *IEEE ICITST*, 2016.

[79] S. Khalifa, G. Lan, M. Hassan, A. Seneviratne, and S. K. Das, "HARKE: Human Activity Recognition from Kinetic Energy Harvesting Data in Wearable Devices," *IEEE Transactions on Mobile Computing*, 2017.

[80] G. De Leonardis, S. Rosati, G. Balestra, *et al.*, "Human Activity Recognition by Wearable Sensors: Comparison of Different Classifiers for Real-Time Applications," *IEEE MeMeA*, 2018.

[81] W. Lu, F. Fan, J. Chu, P. Jing, and S. Yuting, "Wearable Computing for Internet of Things: A Discriminant Approach for Human Activity Recognition," *IEEE Internet of Things Journal*, 2018.

[82] M. Zubair, K. Song, and C. Yoon, "Human Activity Recognition Using Wearable Accelerometer Sensors," *IEEE ICCE-Asia*, 2016.

[83] K.-C. Liu, C.-Y. Yen, L.-H. Chang, C.-Y. Hsieh, and C.-T. Chan, "Wearable Sensor-based Activity Recognition for Housekeeping Task," *IEEE International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, 2017.

[84] A. R. M. Forkan, F. Montori, D. Georgakopoulos, P. P. Jayaraman, A. Yavari, and A. Morshed, "An Industrial IoT Solution for Evaluating Workers' Performance Via Activity Recognition," *IEEE ICDCS*, 2019.

[85] A. Jukan, X. Masip-Bruin, and N. Amla, “Smart Computing and Sensing Technologies for Animal Welfare: A Systematic Review,” *ACM Computing Surveys (CSUR)*, 2017.

[86] L. Duran Lopez, D. Gutierrez Galan, J. P. Dominguez Morales, *et al.*, “A Low-power, Reachable, Wearable and Intelligent IoT Device for Animal Activity Monitoring,” *International Joint Conference on Computational Intelligence*, 2019.

[87] K. M. Ramokapane, D. van der Linden, and A. Zamansky, “Does my Dog Really Need a Gadget? What Can We Learn from Pet Owners’ Amotivations for Using Pet Wearables?” *International Conference on Animal-Computer Interaction*, 2019.

[88] A. Nk, G. Bhat, J. Park, H. G. Lee, and U. Y. Ogras, “Sensor-Classifier Co-Optimization for Wearable Human Activity Recognition Applications,” *IEEE ICESS*, 2019.

[89] A. Jafari, A. Ganesan, C. S. K. Thalisetty, V. Sivasubramanian, T. Oates, and T. Mohsenin, “Sensornet: A Scalable and Low-Power Deep Convolutional Neural Network for Multimodal Data Classification,” *IEEE Transactions on Circuits and Systems*, 2018.

[90] A. U. N. S.N., A. R. Lua, and L. Gonzalez, “PEAT, How Much Am I Burning?” *ACM Multimedia Systems Conference*, 2018.

[91] Y. Yang, Y. Geng, L. Qiu, W. Hu, and G. Cao, “Context-aware Task Offloading for Wearable Devices,” *IEEE ICCCN*, 2017.

[92] R. Fallahzadeh, J. P. Ortiz, and H. Ghasemzadeh, “Adaptive Compressed Sensing at the Fingertip of Internet-of-Things Sensors: An Ultra-Low Power Activity Recognition,” *IEEE DATE*, 2017.

[93] X. Fafoutis, B. Janko, E. Mellios, *et al.*, “SPW-1: A Low-Maintenance Wearable Activity Tracker for Residential Monitoring and Healthcare Applications,” *eHealth*, 2017.

[94] E. Al Safadi, F. Mohammad, D. Iyer, B. J. Smiley, and N. K. Jain, “Generalized Activity Recognition Using Accelerometer in Wearable Devices for IoT Applications,” *IEEE AVSS*, 2016.

[95] J. A. M. Calero, R. Marino, J. M. Lanza-Gutierrez, T. Riesgo, M. Garcia-Valderas, and C. Lopez-Ongil, "Embedded Emotion Recognition within Cyber-Physical Systems Using Physiological Signals," *IEEE DCIS*, 2018.

[96] L. Rui, Y. Yang, Z. Gao, and X. Qiu, "Computation Offloading in a Mobile Edge Communication Network: A Joint Transmission Delay and Energy Consumption Dynamic Awareness Mechanism," *IEEE Internet of Things Journal*, 2019.

[97] D. Iyer, F. Mohammad, Y. Guo, *et al.*, "Generalized Hand Gesture Recognition for Wearable Devices in IoT: Application and Implementation Challenges," *International Conference on Machine Learning and Data Mining in Pattern Recognition*, 2016.

[98] M. Janidarmian, A. Roshan Fekr, K. Radecka, and Z. Zilic, "A Comprehensive Analysis on Wearable Acceleration Sensors in Human Activity Recognition," *Sensors*, 2017.

[99] F. Li, K. Shirahama, M. A. Nisar, L. Köping, and M. Grzegorzek, "Comparison of Feature Learning Methods for Human Activity Recognition Using Wearable Sensors," *Sensors*, 2018.

[100] B. Jagadish, M. S. Kiran, and P. Rajalakshmi, "A Novel System Architecture for Brain Controlled IoT Enabled Environments," *IEEE Healthcom*, 2017.

[101] L. Bergamini, G. Corbellini, and S. Mangold, "Resource-constrained Medium Access Control Protocol for Wearable Devices," *IEEE WiMob*, 2014.

[102] H.-J. Hong, C.-L. Fan, Y.-C. Lin, and C.-H. Hsu, "Optimizing Cloud-based Video Crowdsensing," *IEEE Internet of Things Journal*, 2016.

[103] S. Lemey, S. Agneessens, P. Van Torre, K. Baes, J. Vanfleteren, and H. Rogier, "Wearable Flexible Lightweight Modular RFID Tag with Integrated Energy Harvester," *IEEE Transactions on Microwave Theory and Techniques*, 2016.

[104] F. Salamone, L. Belussi, C. Currò, *et al.*, "Integrated Method for personal Thermal Comfort Assessment and Optimization through Users' Feedback, IoT and Machine Learning: A Case Study," *Sensors*, 2018.

[105] D. Balsamo, G. V. Merrett, B. Zaghari, *et al.*, "Wearable and Autonomous Computing for Future Smart Cities: Open Challenges," *IEEE SoftCOM*, 2017.

[106] D. Kyriazis, T. Varvarigou, D. White, A. Rossi, and J. Cooper, "Sustainable Smart City IoT Applications: Heat and Electricity Management & Eco-conscious Cruise Control for Public Transportation," *IEEE WoWMoM*, 2013.

[107] O. Galinina, K. Mikhaylov, K. Huang, S. Andreev, and Y. Koucheryavy, "Wirelessly Powered Urban Crowd Sensing over Wearables: Trading Energy for Data," *IEEE Wireless Communications*, 2018.

[108] M. Nakhkash, A. Kanduri, A. M. Rahmani, and P. Liljeberg, "End-to-End Approximation for Characterizing Energy Efficiency of IoT Applications," *IEEE NORCAS*, 2019.

[109] M. Golkarifard, J. Yang, Z. Huang, A. Movaghar, and P. Hui, "Dandelion: A Unified Code Offloading System for Wearable Computing," *IEEE Transactions on Mobile Computing*, 2018.

[110] C. Fiandrino, N. Allio, D. Kliazovich, P. Giaccone, and P. Bouvry, "Profiling Performance of Application Partitioning for Wearable Devices in Mobile Cloud and Fog Computing," *IEEE Access*, 2019.

[111] Z. Song, S. Chadha, A. Byalik, and E. Tilevich, "Programming Support for Sharing Resources Across Heterogeneous Mobile Devices," *International Conference on Mobile Software Engineering and Systems*, 2018.

[112] D. Giovanelli, B. Milosevic, D. Brunelli, and E. Farella, "Enhancing Bluetooth Low Energy with Wake-up Radios for IoT Applications," *IEEE IWCMC*, 2017.

[113] T. Lee, J. Han, M.-S. Lee, H.-S. Kim, and S. Bahk, "CABLE: Connection Interval Adaptation for BLE in Dynamic Wireless Environments," *IEEE SECON*, 2017.

[114] R. Snader, R. Kravets, and A. F. Harris III, "Cryptocop: Lightweight, Energy-Efficient Encryption and Privacy for Wearable Devices," *Workshop on Wearable Systems and Applications*, 2016.

[115] Z. Guo, I. G. Harris, C. B. Harris, Y. Jiang, and L.-f. Tsaur, "A Residual Battery-aware Routing Algorithm Based on DSR for BLE Sensor Networks," *IEEE WTS*, 2016.

[116] G. Corbellini, S. Schmid, and S. Mangold, "Two-way Communication Protocol Using Bluetooth Low Energy Advertisement Frames," *International Workshop on Experiences with the Design and Implementation of Smart Objects*, 2015.

[117] M. Xu, F. Qian, M. Zhu, F. Huang, S. Pushp, and X. Liu, "Deepwear: Adaptive Local Offloading for on-Wearable Deep Learning," *IEEE Transactions on Mobile Computing*, 2019.

[118] *MATLAB*, [Online] Available: https://www.mathworks.com/products/matlab.html (Accessed 2024-03-13).

[119] *NS2*, [Online] Available: http://nsnam.sourceforge.net/wiki/index.php/User_Information (Accessed 2024-03-13).

[120] *OPNET*, [Online] Available: http://cms-api.riverbed.com/portal/community_home (Accessed 2024-03-13).

[121] *OMNETPP*, [Online] Available: https://omnetpp.org/ (Accessed 2024-03-13).

[122] *CASTALIA*, [Online] Available: http://omnetsimulator.com/castlia-installation-manual/ (Accessed 2024-03-13).

[123] C. Ragona, F. Granelli, C. Fiandrino, D. Kliazovich, and P. Bouvry, "Energy-Efficient Computation Offloading for Wearable Devices and Smartphones in Mobile Cloud Computing," *IEEE GLOBECOM*, 2015.

[124] Z. Cheng, P. Li, J. Wang, and S. Guo, "Just-in-Time Code Offloading for Wearable Computing," *IEEE Transactions on Emerging Topics in Computing*, 2015.

[125] P. Schulz, M. Matthe, H. Klessig, *et al.*, "Latency Critical IoT Applications in 5G: Perspective on the Design of Radio Interface and Network Architecture," *IEEE Communications Magazine*, 2017.

[126] H. Chen, R. Abbas, P. Cheng, *et al.*, "Ultra-Reliable Low Latency Cellular Networks: Use Cases, Challenges and Approaches," *IEEE Communications Magazine*, 2018.

[127] R. K. Naha, S. Garg, D. Georgakopoulos, *et al.*, "Fog computing: Survey of trends, architectures, requirements, and research directions," *IEEE Access*, 2018.

[128] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the Edge: A Scalable IoT Architecture Based on Transparent Computing," *IEEE Network*, 2017.

[129] G. Premsankar, M. Di Francesco, and T. Taleb, "Edge Computing for the Internet of Things: A Case Study," *IEEE Internet of Things Journal*, 2018.

[130] Z. Hao, E. Novak, S. Yi, and Q. Li, "Challenges and Software Architecture for Fog Computing," *IEEE Internet Computing*, 2017.

[131] M. Collotta, G. Pau, T. Talty, and O. K. Tonguz, "Bluetooth 5: A Concrete Step Forward toward the IoT," *IEEE Communications Magazine*, 2018.

[132] B. Maag, Z. Zhou, and L. Thiele, "W-air: Enabling Personal Air Pollution Monitoring on Wearables," *ACM Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2018.

[133] M. Caria, J. Schudrowitz, A. Jukan, and N. Kemper, "Smart Farm Computing Systems for Animal Welfare Monitoring," *IEEE International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2017.

[134] X. Liu, Y. Zheng, M. W. Phyu, B. Zhao, M. Je, and X. Yuan, "Multiple Functional ECG Signal is Processing for Wearable Applications of Long-Term Cardiac Monitoring," *IEEE Transactions on Biomedical Engineering*, 2010.

[135] T. Wu, J.-M. Redouté, and M. R. Yuce, "A Wireless Implantable Sensor Design with Subcutaneous Energy Harvesting for Long-Term IoT Healthcare Applications," *IEEE Access*, 2018.

[136] C. Savaglio, P. Pace, G. Aloi, A. Liotta, and G. Fortino, "Lightweight Reinforcement Learning for Energy Efficient Communications in Wireless Sensor Networks," *IEEE Access*, 2019.

[137] W. Wang, P. Xu, and L. T. Yang, "Secure Data Collection, Storage and Access in Cloud-Assisted IoT," *IEEE Cloud Computing*, 2018.

[138] O. Akrivopoulos, I. Chatzigiannakis, C. Tselios, and A. Antoniou, "On the Deployment of Healthcare Applications over Fog Computing Infrastructure," *IEEE COMPSAC*, 2017.

[139] J. Yan, M. Zhou, and Z. Ding, "Recent Advances in Energy-Efficient Routing Protocols for Wireless Sensor Networks: A Review," *IEEE Access*, 2016.

[140] H. B. Elhadj, L. Chaari, and L. Kamoun, "A Survey of Routing Protocols in Wireless Body Area Networks for Healthcare Applications," *International Journal of E-Health and Medical Communications (IJEHMC)*, 2012.

[141] S. Sojuyigbe and K. Daniel, "Wearables/IoT Devices: Challenges and Solutions to Integration of Miniature Antennas in Close Proximity to the Human Body," *IEEE Symposium on Electromagnetic Compatibility and Signal Integrity*, 2015.

[142] Y. Lim, S. Daas, S. Hashim, R. Sidek, N. Kamsani, and F. Rokhani, "Reduced Hardware Architecture for Energy-Efficient IoT Healthcare Sensor Nodes," *IEEE ICSyS*, 2015.

[143] K. Raja, S. Saravanan, R. Anitha, S. S. Priya, and R. Subhashini, "Design of a Low Power ECG Signal Processor for Wearable Health System-Review and Implementation Issues," *IEEE ISCO*, 2017.

[144] A. J. Casson and E. Rodriguez-Villegas, "Low Power Signal Processing Electronics for Wearable Medical Devices," *International Conference of the IEEE Engineering in Medicine and Biology*, 2010.

[145] G. Yang, J. Chen, F. Jonsson, H. Tenhunen, and L.-R. Zheng, "An ASIC Solution for Intelligent Electrodes and Active-Cable used in a Wearable ECG Monitoring System," *BIODEVICES*, 2009.

[146] M. Magno, S. Marinkovic, D. Brunelli, L. Benini, and E. Popovici, "Combined Methods to Extend the Lifetime of Power Hungry WSN with Multimodal Sensors and Nanopower Wakeups," *International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2012.

[147] M. Amadeo, C. Campolo, and A. Molinaro, "Multi-Source Data Retrieval in IoT via Named Data Networking," *ACM Conference on Information-Centric Networking*, 2014.

[148] Z. Chi, Y. Li, Y. Yao, and T. Zhu, "PMC: Parallel Multi-Protocol Communication to Heterogeneous IoT Radios within a Single WiFi Channel," *IEEE ICNP*, 2017.

[149] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra Low Power Wake-up Radios: A Hardware and Networking Survey," *IEEE Communications Surveys & Tutorials*, 2017.

[150] P. Bulić, G. Kojek, and A. Biasizzo, "Data Transmission Efficiency in Bluetooth Low Energy versions," *Sensors*, 2019.

[151] V. Deep and T. Elarabi, "Efficient IEEE 802.15. 4 ZigBee Standard Hardware Design for IoT Applications," *IEEE ICSigSys*, 2017.

[152] S. Warren and B. Natarajan, "Wireless Communication Technologies for Wearable Systems," *Wearable Monitoring Systems*, 2011.

[153] M. M. Alam, H. Malik, M. I. Khan, T. Pardy, A. Kuusik, and Y. Le Moullec, "A Survey on the Roles of Communication Technologies in IoT-based Personalized Healthcare Applications," *IEEE Access*, 2018.

[154] Y.-Y. Ting and F. J. Lin, "A Comparison and Evaluation of Different BLE Connection Methods for Wearable Devices," *IEEE CSCN*, 2017.

[155] A. J. Jara, D. Fern'ndez, P. López, M. A. Zamora, B. Ubeda, and A. G. Skarmeta, "Evaluation of Bluetooth Low Energy Capabilities for Continuous Data Transmission from a Wearable Electrocardiogram," *IEEE International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, 2012.

[156] *LoRa Alliance*, [Online] Available: https://www.lora-alliance.org/ (Accessed 2024-03-13).

[157] *Sigfox*, [Online] Available: https://www.sigfox.com (Accessed 2024-03-13).

[158] A. Ometov, N. Daneshfar, A. Hazmi, *et al.*, "System-Level Analysis of IEEE 802.11ah Technology for Unsaturated MTC Traffic," *International Journal of Sensor Networks*, 2018.

[159] O. Apilo, J. Mäkelä, and A. Kuosmonen, "Evaluation of Cellular IoT for Sport Wearables," *IEEE PIMRC*, 2019.

[160] K. C. Barr and K. Asanović, "Energy-Aware Lossless Data Compression," *ACM TOCS*, 2006.

[161] S. Xiao, A. Dhamdhere, V. Sivaraman, and A. Burdett, "Transmission Power Control in Body Area Sensor Networks for Healthcare Monitoring," *IEEE Journal on Selected Areas in Communications*, 2009.

[162] S. Lin, F. Miao, J. Zhang, *et al.*, "ATPC: Adaptive Transmission Power Control for Wireless Sensor Networks," *ACM TOSN*, 2016.

[163] A. H. Sodhro, Y. Li, and M. A. Shah, "Energy-Efficient Adaptive Transmission Power Control for Wireless Body Area Networks," *IET Communications*, 2016.

[164] B. H. Jung, H. Jin, and D. K. Sung, "Adaptive Transmission Power Control and Rate Selection Scheme for Maximizing Energy Efficiency of IEEE 802.11 Stations," *IEEE PIMRC*, 2012.

[165] P. Pace, G. Fortino, Y. Zhang, and A. Liotta, "Intelligence at the Edge of Complex Networks: The Case of Cognitive Transmission Power Control," *IEEE Wireless Communications*, 2019.

[166] S. Padalkar, A. Korlekar, and U. Pacharaney, "Data Gathering in Wireless Sensor Network for Energy Efficiency with and without Compressive Sensing at Sensor Node," *IEEE ICCSP*, 2016.

[167] A. Fragkiadakis, P. Charalampidis, and E. Tragos, "Adaptive Compressive Sensing for Energy Efficient Smart Objects in IoT Applications," *IEEE VITAE*, 2014.

[168] H. Mamaghanian, N. Khaled, D. Atienza, and P. Vandergheynst, "Compressed Sensing for Real-time Energy-Efficient ECG Compression on Wireless Body Sensor Nodes," *IEEE Transactions on Biomedical Engineering*, 2011.

[169] D. E. Bellasi and L. Benini, "Energy-Efficiency Analysis of Analog and Digital Compressive Sensing in Wireless Sensors," *IEEE Transactions on Circuits and Systems*, 2015.

[170] Y. Zhang, Y. Xiang, L. Y. Zhang, Y. Rong, and S. Guo, "Secure Wireless Communications Based on Compressive Sensing: A Survey," *IEEE Communications Surveys & Tutorials*, 2018.

[171] S. K. Sharma, E. Lagunas, S. Chatzinotas, and B. Ottersten, "Application of compressive sensing in cognitive radio communications: A survey," *IEEE Communications Surveys & Tutorials*, 2016.

[172] C. Doukas and I. Maglogiannis, "Managing Wearable Sensor Data through Cloud Computing," *International Conference on Cloud Computing Technology and Science*, 2011.

[173] M. A. Razzaque, C. Bleakley, and S. Dobson, "Compression in Wireless Sensor Networks: A Survey and Comparative Evaluation," *ACM TOSN*, 2013.

[174] H. Huang, S. Hu, and Y. Sun, "Energy-Efficient ECG Compression in Wearable Body Sensor Network by Leveraging Empirical Mode Decomposition," *IEEE EMBS (BHI)*, 2018.

[175] S. Pushpalatha and K. Shivaprakasha, "Energy-Efficient Communication Using Data Aggregation and Data Compression Techniques in Wireless Sensor Networks: A Survey," *Advances in Communication, Signal Processing, VLSI, and Embedded Systems*, 2020.

[176] N. Kimura and S. Latifi, "A Survey on Data Compression in Wireless Sensor Networks," *IEEE International Conference on Information Technology: Coding and Computing*, 2005.

[177] P. Kavitha, "A Survey on Lossless and Lossy Data Compression Methods," *International Journal of Computer Science & Engineering Technology (IJCSET)*, 2016.

[178] Q. Xu, T. Mytkowicz, and N. S. Kim, "Approximate Computing: A Survey," *IEEE Design & Test*, 2015.

[179] P. Stanley-Marbell, A. Alaghi, M. Carbin, *et al.*, "Exploiting Errors for Efficiency: A Survey from Circuits to Algorithms," *ACM Computing Surveys (CSUR)*, 2020.

[180] G. Wilson, J. Bryan, K. Cranston, J. Kitzes, L. Nederbragt, and T. K. Teal, "Good Enough Practices in Scientific Computing," *PLoS Computational Biology*, 2017.

[181] S. Mittal, "A Survey of Techniques for Approximate Computing," *ACM Computing Surveys (CSUR)*, 2016.

[182] A. Ometov, P. Masek, L. Malina, *et al.*, "Feasibility Characterization of Cryptographic Primitives for Constrained (Wearable) IoT Devices," *IEEE PerCom*, 2016.

[183] A. Ometov, Y. Bardinova, A. Afanasyeva, *et al.*, "An Overview on Blockchain for Smartphones: State-of-the-Art, Consensus, Implementation, Challenges and Future Trends," *IEEE Access*, 2020.

[184] Y. Bardinova, K. Zhidanov, S. Bezzateev, M. Komarov, and A. Ometov, "Measurements of Mobile Blockchain Execution Impact on Smartphone Battery," *Data*, 2020.

[185] O. Arias, J. Wurm, K. Hoang, and Y. Jin, "Privacy and Security in Internet of Things and Wearable Devices," *IEEE Transactions on Multi-Scale Computing Systems*, 2015.

[186] "Gartner, Global End-User Spending on Wearable Devices to Total $81.5 Billion in 2021.,"

[187] X. Lyu, H. Tian, C. Sengul, and P. Zhang, "Multiuser Joint Task Offloading and Resource Optimization in Proximate Clouds," *IEEE Transactions on Vehicular Technology*, 2016.

[188] C.-F. Liu, M. Bennis, M. Debbah, and H. V. Poor, "Dynamic Task Offloading and Resource Allocation for Ultra-Reliable Low-Latency Edge Computing," *IEEE Transactions on Communications*, 2019.

[189] *Google Glass: Tech specs*, [Online] Available: https://support.google.com/glass/answer/3064128?hl=en (Accessed 2024-03-13).

[190] W. B. Qaim, A. Ometov, C. Campolo, A. Molinaro, E. S. Lohan, and J. Nurmi, "Understanding the performance of task offloading for wearables in a two-tier edge architecture," *IEEE ICUMT*, 2021.

[191] X. Lyu and H. Tian, "Adaptive Receding Horizon Offloading Strategy Under Dynamic Environment," *IEEE Communications Letters*, 2016.

[192] M. Othman, S. A. Madani, S. U. Khan, *et al.*, "A Survey of Mobile Cloud Computing Application Models," *IEEE Communications Surveys & Tutorials*, 2013.

[193] S. Misra, B. E. Wolfinger, M. Achuthananda, T. Chakraborty, S. N. Das, and S. Das, "Auction-based Optimal Task Offloading in Mobile Cloud Computing," *IEEE Systems Journal*, 2019.

[194] W. Zhang, Y. Wen, and D. O. Wu, "Energy-Efficient Scheduling Policy for Collaborative Execution in Mobile Cloud Computing," *IEEE INFOCOM*, 2013.

[195] T. D. Burd and R. W. Brodersen, "Processor Design for Portable Systems," *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology*, 1996.

[196] Y. Jang, J. Na, S. Jeong, and J. Kang, "Energy-Efficient Task Offloading for Vehicular Edge Computing: Joint Optimization of Offloading and Bit Allocation," *IEEE VTC2020-Spring*, 2020.

[197] *802.11 OFDM Data Rates – The Math Behind The Numbers*, [Online] Available: https://dot11.exposed/2018/11/29/802-11-ofdm-data-rates-the-math-behind-the-numbers/ (Accessed 2024-03-13).

[198] D. Halperin, B. Greenstein, A. Sheth, and D. Wetherall, "Demystifying 802.11n Power Consumption," *International Conference on Power Aware Computing and Systems*, 2010.

[199] S. Deb and P. Monogioudis, "Learning-based Uplink Interference Management in 4G LTE Cellular Systems," *IEEE/ACM Transactions on Networking*, 2014.

[200] 3GPP TR 36.814, *Evolved Universal Terrestrial Radio Access (E-UTRA); Further Advancements for E-UTRA Physical Layer Aspects (Release 9)*, 2017.

[201] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation Rate Maximization in UAV-enabled Wireless-Powered Mobile-Edge Computing Systems," *IEEE Journal on Selected Areas in Communications*, 2018.

[202] K. Cheng, Y. Teng, W. Sun, A. Liu, and X. Wang, "Energy-Efficient Joint Offloading and Wireless Resource Allocation Strategy in Multi-MEC Server Systems," *IEEE ICC*, 2018.

[203] 3GPP TR 36.101, *Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) Radio Transmission and Reception (Release 8)*, 2021.

[204] *OnePlus Nord CE 5G Specifications*, [Online] Available: https://www.oneplus.com/uk/nord-ce-5g/specs (Accessed 2024-03-13).

[205] N. Jain, X. Fan, W. D. Leon-Salas, and A. M. Lucietto, "Extending Battery Life of Smartphones by Overcoming Idle Power Consumption Using Ambient Light Energy Harvesting," *IEEE ICIT*, 2018.

[206] D. Ferreira, C. Schuss, C. Luo, J. Goncalves, V. Kostakos, and T. Rahkonen, "Indoor Light Scavenging on Smartphones," *International Conference on Mobile and Ubiquitous Multimedia*, 2016.

[207] A. Carroll, G. Heiser, *et al.*, "An Analysis of Power Consumption in a Smartphone," *USENIX Annual Technical Conference*, 2010.

[208] R. LiKamWa, Z. Wang, A. Carroll, F. X. Lin, and L. Zhong, "Draining Our Glass: An Energy and Heat Characterization of Google Glass," *Asia-Pacific Workshop on Systems*, 2014.

[209] B. Wang, C. Wang, W. Huang, Y. Song, and X. Qin, "A Survey and Taxonomy on Task Offloading for Edge-Cloud Computing," *IEEE Access*, 2020.

[210] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: State-of-the-art and research challenges," *Journal of Internet Services and Applications*, 2010.

[211] O. Khalid, M. U. S. Khan, S. U. Khan, and A. Y. Zomaya, "OmniSuggest: A Ubiquitous Cloud-based Context-Aware Recommendation System for Mobile Social Networks," *IEEE Transactions on Services Computing*, 2013.

[212] K. Goel and M. Goel, "Cloud Computing Based E-Commerce Model," *IEEE RTEICT*, 2016.

[213] S. T. Siddiqui, S. Alam, Z. A. Khan, and A. Gupta, "Cloud-based E-Learning: Using Cloud Computing Platform for an Effective E-Learning," *Smart Innovations in Communication and Computational Sciences*, 2019.

[214] M. Marjani, F. Nasaruddin, A. Gani, *et al.*, "Big IoT Data Analytics: Architecture, Opportunities, and Open Research Challenges," *IEEE Access*, 2017.

[215] C. Pritchard, Y. Beheshti, and M. Sepahi, *Mobile Edge Computing: Architecture, Use-cases, Applications*. 2020.

[216] A. Robles-Enciso and A. F. Skarmeta, "A multi-layer guided reinforcement learning-based tasks offloading in edge computing," *Computer Networks*, vol. 220, p. 109 476, 2023.

[217] F. Saeik, M. Avgeris, D. Spatharakis, *et al.*, "Task offloading in edge and cloud computing: A survey on mathematical, artificial intelligence and control theory solutions," *Computer Networks*, vol. 195, p. 108 177, 2021.

[218] Z. Gao, W. Hao, Z. Han, and S. Yang, "Q-Learning-Based Task Offloading and Resources Optimization for a Collaborative Computing System," *IEEE Access*, 2020.

[219] X. Ma, C. Yin, and X. Liu, "Machine Learning Based Joint Offloading and Trajectory Design in UAV Based MEC System for IoT Devices," *IEEE ICCC*, 2020.

[220] F. Jiang, W. Liu, J. Wang, and X. Liu, "Q-Learning Based Task Offloading and Resource Allocation Scheme for Internet of Vehicles," *IEEE ICCC*, 2020.

[221] M. Min, L. Xiao, Y. Chen, P. Cheng, D. Wu, and W. Zhuang, "Learning-based Computation Offloading for IoT Devices with Energy Harvesting," *IEEE Transactions on Vehicular Technology*, 2019.

[222] Y. Li, L. Chen, D. Zeng, and L. Gu, "A customized reinforcement learning based binary offloading in edge cloud," *IEEE ICPADS*, 2020.

[223] K. Jiang, H. Zhou, D. Li, X. Liu, and S. Xu, "A Q-learning based Method for Energy-Efficient Computation Offloading in Mobile Edge Computing," *IEEE ICCCN*, 2020.

[224] A. Shakarami, M. Ghobaei-Arani, and A. Shahidinejad, "A Survey on the Computation Offloading Approaches in Mobile Edge Computing: A Machine Learning-based Perspective," *Computer Networks*, 2020.

[225] Y. Liu, D. Lu, G. Zhang, J. Tian, and W. Xu, "Q-learning based content placement method for dynamic cloud content delivery networks," *IEEE Access*, 2019.

[226] J. Zhang, X. Hu, Z. Ning, *et al.*, "Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks," *IEEE Internet of Things Journal*, 2017.

[227] K. Zhang, X. Gui, D. Ren, and D. Li, "Energy–latency tradeoff for computation offloading in uav-assisted multiaccess edge computing system," *IEEE Internet of Things Journal*, 2020.

[228] O.-K. Shahryari, H. Pedram, V. Khajehvand, and M. D. TakhtFooladi, "Energy and task completion time trade-off for task offloading in fog-enabled iot networks," *Pervasive and Mobile Computing*, 2021.

[229] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[230] F. Liu, Z. Huang, and L. Wang, "Energy-efficient collaborative task computation offloading in cloud-assisted edge computing for iot sensors," *Sensors*, 2019.

[231] J. Du, L. Zhao, J. Feng, and X. Chu, "Computation offloading and resource allocation in mixed fog/cloud computing systems with min-max fairness guarantee," *IEEE Transactions on Communications*, 2017.

[232] P. Di Lorenzo, S. Barbarossa, and S. Sardellitti, "Joint Optimization of Radio Resources and Code Partitioning in Mobile Edge Computing," *arXiv preprint arXiv:1307.3835*, 2013.

[233] *The NS-3 Network Simulator*. [Online] Available: http://www.nsnam.org/ (Accessed 2024-03-13).

[234] B. Shi, J. Yang, Z. Huang, and P. Hui, "Offloading guidelines for augmented reality applications on wearable devices," *ACM international conference on Multimedia*, 2015.

[235] *2.4/5GHz Dual-band 1x1 Wi-Fi 5 (802.11ac) and Bluetooth 5.2 Solution*, [Online] Available: https://www.nxp.com/docs/en/data-sheet/88W8987_SDS.pdf (Accessed 2024-03-13).

[236] *ns3 user documentation*, *using the WifiNetDevice*, [Online] Available: https://www.nsnam.org/docs/models/html/wifi-user/using-the-wifinetdevice (Accessed 2024-03-13).

[237] B. Jang, M. Kim, G. Harerimana, and J. W. Kim, "Q-learning algorithms: A comprehensive classification and applications," *IEEE access*, vol. 7, pp. 133 653–133 667, 2019.