Title
Perpetual Operations in IoT Deployments for Smart Communities

Author
Alhassoun, Nailah

Publication Date
2021

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA,
IRVINE

Perpetual Operations in IoT Deployments for Smart Communities

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computer Science

by

Nailah Alhassoun

Dissertation Committee:
Professor Nalini Venkatasubramanian, Chair
Professor Nikil Dutt
Professor Ramesh Jain
Professor Sharad Mehrotra

2021

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First and foremost, I thank Almighty Allah for giving me the strength and patience to work through all these years so that today I can stand proudly with my head held high for this achievement.

I would like to start by acknowledging my Ph.D. advisor Professor Nalini Venkatasubramanian, for her guidance, support, encouragement, patience, and her believe in me and my work throughout my research period. I also would like to thank the committee members of my dissertation, Professor Nikil Dutt, Professor Ramesh Jain, Professor Sharad Mehrotra for their valuable feedback and devoting time.

I would like to thank my coauthors on the publications especially professor Md Yusuf Sarwar Uddin. I also extend my thanks to my colleagues in the Distributed Systems Middleware group for sharing their expertise and knowledge.

Last but not least, I would like to express my deepest gratitude toward my family for their support and love. I deeply thank my parents, Saleh and Layla, for their prayers, unconditional love and support. Also, I will be forever thankful to my husband Tawfiq for his love, support, understanding, and encouragement through hectic times.

VITA

Nailah Alhassoun

EDUCATION

Doctor of Philosophy in Computer Science **2021**
University of California Irvine *Irvine, California*

Master of Science in Computer Science
University of California Irvine *Irvine, California*

Master of Science in Computer Science
King Saud University *Riyadh, Saudi Arabia*

Bachelor of Science in Computer Science
King Saud University *Riyadh, Saudi Arabia*

RESEARCH EXPERIENCE

Graduate Research Assistant **2015–2021**
University of California Irvine *Irvine, California*

TEACHING EXPERIENCE

Teaching Assistant **2017–2020**
University of California Irvine *Irvine, California*

Lecturer **2010–2013**
King Saud University *Riyadh, Saudi Arabia*

REFEREED PUBLICATIONS

Multi-Network Provisioning for Perpetual Operations in IoT-Enabled Smart Spaces **SMARTCOMP, 2020**
IEEE International Conference on Smart Computing

Cross-Layer Energy Optimization for IoT Platforms **PerCom, 2020**
IEEE International Conference on Pervasive Computing and Communications

Context-Aware Energy Optimization for Perpetual IoT-Based Safe Communities **SUSCOM, 2019**
Sustainable Computing: Informatics and Systems

SAFER: An IoT-Based Perpetual Safe Community Awareness and Alerting Network **IGSC, 2017**
IEEE International Green and Sustainable Computing Conference

The SCALE2 Multi-network Architecture for IoT-based Resilient Communities **SMARTCOMP, 2016**
IEEE International Conference on Smart Computing

SOFTWARE

SAFER `https://scale.ics.uci.edu`
IoT system; deployed in homes to detect critical events (injury, hazardous-environment) that must trigger immediate action and response.

IoT Energy Simulator
Python simulator that simulate activities and apply algorithms to optimize system energy.

ABSTRACT OF THE DISSERTATION

Perpetual Operations in IoT Deployments for Smart Communities

By

Nailah Alhassoun

Doctor of Philosophy in Computer Science

University of California, Irvine, 2021

Professor Nalini Venkatasubramanian, Chair

The IoT revolution has provided a promising opportunity to build powerful perpetual awareness systems. Perpetual awareness systems are sensing systems characterized by continuous monitoring of spaces, people and events; they are essential to many safety and mission-critical applications, e.g. assisted living, healthcare and public safety. Today, IoT platforms are the key technology substrate for smart homes/buildings that are equipped with heterogeneous devices and diverse (often multiple) network interfaces. Data thus generated can be processed locally or at a cloud to create knowledge for diverse ubiquitous services.

Many end-to-end challenges arise in operating these deployments. The key question in this thesis is: how to ensure perpetual operations in IoT deployments? We address key challenges in perpetual smart-space applications that affect system reliability, lifetime and availability, i.e. that of energy consumption, processing overhead associated with continuous sensing, communication and processing. To understand issues associated with perpetual sensing we developed and deployed a smart assisted living platform in a senior-care facility in Montgomery County, MD to detect critical events (injury, hazardous-environment, elderly falls) that require immediate action and response, where battery-operated and wall-powered IoT devices are co-executed to ensure the safety of occupants.

We observed that diverse applications utilize data at different levels of quality; e.g. different

fall detection applications utilize diverse multi-modal sensory data such as tri-axis accelerometer, video image data, .. etc. Each approach delivers different fall detection accuracy, and uses algorithms with different complexity and memory allocation, so that is why resources consumed for sensing, computation and communication vary based on the desired quality. We exploit these quality tolerances by modeling them as ”space-states” and intelligently leverage the dynamic space-states to select and provision resources (access networks, device capabilities, processing location) to reduce energy and processing overhead while ensuring application quality. Furthermore, diverse people have different levels of needs; we exploit these needs by modeling them as “personal-space-states” and leverage the dynamic workload to reduce processing overhead while ensuring an efficient perpetual IoT system.

In this thesis, we design a prototype SAFER, a novel semantic approach that utilizes context extracted, trigger actions and space-states shifts for each user to drive energy-optimized sensor/network/processing activations. To validate our approach, we derive use cases from real-world assisted living smarthomes with multiple personal and in-situ devices for targeting applications such as elderly fall detection. Through detailed testbed measurements and larger simulated scenarios, we show that our provisioning algorithms and adaptive techniques that use semantics can achieve reductions in energy dissipation, active devices and maximizing system lifetime without loss of application quality/accuracy.

Chapter 1

Introduction

In this chapter, we introduce the Internet of Things (IoT) and provide a summary of community IoT deployments examples. Then, we show how these IoT deployments and smart spaces have enabled perpetual/continuous monitoring in multi scales and contexts, from personal scale such as monitoring vital signs and activities of individuals to city-scale monitoring for water systems and traffic.

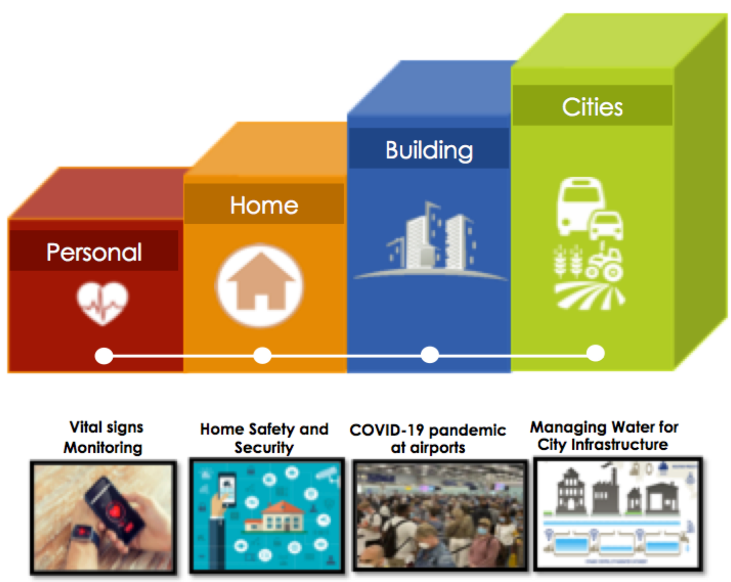


Figure 1.1: IoT-Enabled Continuous/Perpetual Monitoring in Multi Scale, Context

SAFER, an assisted living platform, serves as a real-world usecase to motivate research challenges and serve as a testbed for deploying our proposed research techniques. In this perpetual sensing usecase, we leverage both mobile and in-situ IoT deployments in communities and aim to understand how multiple sensing and communication modes can ensure perpetual operations in IoT deployments.

In particular, we strive to fully exploit the potential of the mobile, in-situ, battery-operated and wall-powered IoT deployments in community IoT systems by appropriately leveraging semantic data (e.g. activity of daily living, time, location, etc.), along with data from IoT infrastructures (e.g. device battery profile, available network interfaces, etc.) to meet application requirements (e.g. latency, accuracy, etc.).

1.1 Internet of Things (IoT) Revolution

Recent advances in device, communication and processing technologies have created the Internet of Things (IoT) revolution; this is considered to be a key technological enabler that can embed intelligence into various aspects of our daily lives by creating smart homes, communities, and city level infrastructures worldwide. This growing in Internet of Things (IoT) ecosystem with over 50 billion connected devices is accelerating the pace of IoT deployments to offer diverse services in everyday living spaces.

For instance, smart buildings, today, contain heterogeneous IoT devices, that manage control systems to improve the safety and comfort of building occupants by enabling a range of applications such as surveillance/security and fire protection while reducing energy footprints to lower operational cost.

Similarly, in the past decade, smart home sensing systems have created the capability to automatically and unobtrusively collect information about a resident's everyday behavior and provide value-added services derived from this information. Such efforts have used sensing technology to monitor the resident's activities in their home environment [55]. The new generation of personal sensing devices has also created the ability to sense personal health factors (e.g. heart rate, activity) and is making possible always connected health-care IoT solutions for elder care and care for individuals with disabilities.

1.2 IoT-Enabled Smart Spaces and Perpetual Monitoring

These IoT deployments encompass pervasive sensing, intelligent interaction, smart control and created connected places which known as smart spaces. Smart spaces are physical locations equipped with networked sensors to give users more and better information about the condition of those locations and how they're used. Locations for smart spaces can range from a building with networked temperature and motion sensors to a vehicle that constantly reports its location, performance and maintenance needs. Smart spaces combine the power of the Internet of Things (IoT) and analytics to improve the experience, safety, efficiency, and quality of our life; and enable perpetual/continuous monitoring of spaces, people, activities and events.

Perpetual systems are sensing systems that can continuously monitor (24/7) the underlying space and provide key insights and services in the smartspace without interruption. Application domains for perpetual systems can range from societal services such as public safety, assisted living and healthcare, industrial monitoring and control, safe operation of mission-critical facilities (e.g. nuclear power plants) and inventory management for enterprises.
For instance, a home-based healthcare monitoring deployment can capture the resident's Activities of Daily Living(ADLs) to facilitate patient tracking, health assessments and emer-

Figure 1.2: IoT-Enabled Perpetual Systems

gency response services in a manner that is easier, quicker and safer. Such services enable seniors or individuals with disabilities to live independently; it can be used to provide assurance to family members/caregivers who cannot be available to provide care 24/7. Perpetual health monitoring is also useful in nursing care and rehabilitation facilities that are facing a shortage of qualified nurses/doctors to monitor patients with chronic diseases [97, 10, 11, 30]. A rather different usecase is that of a smart campus with IoT-instrumented buildings [3] which offer a novel range of services - smart classrooms/meeting spaces that are able to capture activities and interactions to improve educational outcomes, public safety/environmental monitoring for health/safety, building energy management for sustainability etc.

Consider, next, a smart retail setting with autonomous inventory monitoring[86, 87] and checkout systems in physical retail stores. Studies indicate that the growing cost of human labor is unable to sustain traditional manual approaches to inventory inspection and tagging of items - about 4% of sales are lost due to an average 5-10% out-of-shelf stock-out rate. Experts argue that the design of reliable/accurate automated inventory monitoring and checkout systems can revolutionize physical retail by bringing down operating costs.

1.3 IoT in Mission-critical Settings

IoT-enabled deployments are gaining popularity for mission-critical societal applications that require perpetual operations including lifeline infrastructures for communities such as water, power, transportation and in critical services such as healthcare and assisted living. In such mission-critical systems the devices and platforms must operate reliably and function continuously in extreme conditions, e.g. disasters. Mission-critical applications include surveillance and security to personal safety and situational awareness in emergency response scenarios. One relevant helthcare mission-critical usecase is elderly fall detection, which will be described in the following section. Similarly, a smart home security system can assist those with disabilities (with vision impairments, deaf and hard of hearing) by capturing anomalous events and intrusions and providing alerts in the event of suspicious movement.

1.3.1 Driving Use Case- Elderly Fall Detection System

Falls, espically among older adults, are a major challenge in the public health care domain. According to the U.S. Department of Health and Human Services in 2011, the senior population (65+) represented 13.3% of the U.S. population- this depided an increase from 35 million in 2000 to 41.4 million in 2011. About 28% (11.8 million) of all noninstitutionalized seniors in 2012 lived alone. In addition, according to the Centers for Disease Control and Prevention, one out of three older people fall each year and 2.5 million elderly people are treated in emergency departments for fall injuries.

For the elderly who experience serious fall injuries, the amount of time spent immobile often affects their health outcome. Muscle cell breakdown starts within 30-60 minutes of compression due to falling. Consequently, one of the most important personal sensing systems in the safe communities for elderly people is a fall detection system that will enable rapid triage.

The design of reliable systems to quickly detect and mitigate the effects of falls will help improve outcomes significantly.

Sensing platforms for fall detection can be heterogeneous [72]. Wearable based approaches rely on embedded body-worn sensors to detect the motion, orientation and location of the body e.g. smart pendants and watches. Ambient approaches attempt to fuse data from sensors in the event including audio, pressure sensors for object detection and tracking. Vision based systems use images taken by cameras to model the posture of the body using machine learning techniques.

While each approach provide a known level of accuracy, they also suffer from drawbacks. The ambient sensing system suffer from reliability while the wearable system suffered from limited resources and power consumption. Also, how vision-based approaches such as cameras/microphones may cause privacy violations since they are intrusive and capture personal information. We argue that combining multiple sensors can enhance the performance under real-life conditions by improving the accuracy, reliability, usability, and power consumption.

1.4 Key Challenges

While IoT deployments hold significant promises to improve the quality of life of citizens, several limitations arise in operating IoT deployments in a scalable, resilient manner over time. First, mission-critical awareness systems are expected to operate 24/7, i.e. perpetually, to monitor and detect any critical event. This raises key operational challenges/issues of the cost of operation and continuous energy consumption associated with continuous IoT sensing/communication operations. These challenges arise both at the device scale and in a broader sense, at the systemic level. Second, IoT devices typically are wireless and small in size with restricted resources including limited compute power, battery and storage capability.

Continuous operation implies increasing plug-loads, frequent data uploads and battery replacements. Third, the IoT setting is dynamic - it includes devices with varying capabilities in terms of computation, sensing, mobility and communication; these devices use diverse communication protocols and direct connections to cloud platforms.

Fourth, the need for low cost and mass-scale production further enhances the likelihood of component variability and structural failures. This huge scale of deployment raises questions on the total energy impact when numerous devices must operate perpetually; indeed the carbon footprint caused by the IoT industry is non-trivial.

Lastly, the diversity of settings and deployments play an important role in both the accuracy and cost of the applications deployed.

1.5 Thesis Contributions

Overall, in the context of IoT deployments, this thesis aims to address the above-mentioned challenges through cross-layer optimizing techniques that exploit the semantics and heterogeneity of IoT devices in three different layers: sensing, networking and processing to perpetual IoT system. Key scientific contributions include:

- A holistic mission-critical IoT system design that could leverage the advantages of both battery-powered and wall-powered deployments with diverse capabilities while supporting different IoT application requirements.
- The design and implementation of 'SCALE-personal sensing and SAFER', our assisted-living IoT prototype systems, based on the measurements that we carry out in our real-world test-beds studies.

- The key challenges in the context of perpetual mission-critical IoT deployments that we identify from our fall detection system experience gained in implementation, measurement study, and maintenance.
- The abstraction of space context and state (e.g., ADLs, events) in comprehensive scheduling solutions and dynamic provisioning strategies.
- The idea of efficient management of heterogeneous low-cost IoT devices, networking, processing locations and the activation techniques based on semantics to achieve such reliable perpetual monitoring and extended the platform lifetime.

1.6 Plan of Thesis and Organization

The organization of the thesis is listed as follows. For each level of optimization, we formalize the respective research problem and provide a solution/algorithm to solving it effectively; extensive simulations are based on real and synthetic data, and the results show significant improvement compared to naive approaches.

- Chapter 2 surveys related work.
- Chapter 3 proposes our semantic (SAFER) approach and system architecture in greater detail.
- Chapter 4 describes our three-phase optimization framework. It maintains semantic activation and scheduling to address tradeoffs in energy efficiency and reliable sensing by leveraging (a) Heterogeneity of IoT devices, (b) Context of activities extracted from IoT data (e.g. activities of daily living), (c) Diverse energy profiles of IoT deployment configurations.

- Chapter 5 we extended our approach to the networking layer by utilizing the multiple interfaces and communication options in IoT deployments to reduce communication overhead while ensuring application quality.
- Chapter 6 we scaled up our perpetual operations system for multiple applications/people with diverse needs which raises processing overhead. Therefore, to handle this computational complexity, we utilized edge notion (local processing) and exploiting users semantics as abstracted modes "personal-space-state" to run several applications efficiently and perpetually.
- Chapter 7 concludes the dissertation with lessons learned and looks forward to future research problems we must address to enable our vision of perpetual CPHS systems.

Chapter 2

Related Work

Perpetual monitoring systems are becoming common in multiple contexts (public safety, autonomous inventory, mission-critical, assisted living and healthcare) [10, 11, 97, 3, 30, 86, 87]; however, continuous sensing/transmission and processing for operational purposes in such settings have cross-layer concerns at multiple levels of the system (devices, networks, data, compute etc.). One of the main concerns that gained increasing attention from research communities, is the energy consumption that increases the operation costs and affects the system lifetime; many research efforts and novel techniques have been proposed to reduce energy consumption in sensor networks at the device, communication/network and system levels.

In the IoT ecosystem is characterized by a large level of heterogeneity, dynamicity and horizontal scalability (plug and play on the fly). It includes devices with varying capabilities in terms of computation, sensing, energy source/consumption, mobility and communication; these devices use diverse communication protocols and direct connections to local or cloud platforms. Also, sometimes in IoT settings the node computation load can be higher than the transmission load. Despite the term IoT is relatively new, the idea of monitoring and controlling devices through computers and networks has been around for decades as we mentioned in the previous paragraph. IoT is a new paradigm that integrates several technologies that already existed, such as WSN. While the number of scientific publications related to WSN has been declining in recent years, this is not due to WSN is losing importance, but rather, researchers are beginning to treat WSN as a technology integrated into the IoT ecosystem.

2.1 Energy optimization at the *device level*

Vendors and engineers look into circuit, and hardware optimization to reduce power consumption through dynamic voltage and frequency scaling, power-aware scheduling, power mode management, micro architectural techniques and energy harvesting methods (motion, thermal energy extraction, wireless energy harvesting[15, 50, 85, 36, 26]). Dynamic voltage and frequency scaling (DVFS) is adopted to reduce power consumption by configuring the processor based on the requirements of the executing applications [14].

Moreover, power-aware scheduling aims at maximally exploiting the benefit of power manageable resources through switching among multiple power saving modes as different modes consume different amount of power [79]. Also, micro-architectural techniques leverage application properties to dynamically reconfigure a specific component of the system to save energy, such as main memory ,cache [102, 98, 69, 70].

In addition, energy harvesting methods where energy is derived from external sources, such as vibration, light, thermal, human motion and wireless energy harvesting to prolong battery longevity [50, 36, 45]. In [26] authors introduced a low-cost and efficient renewable energy monitoring system, implements LoRa network without base station, and collects energy status data from solar and wind power generation facilities.

2.2 Energy efficiency approaches *at the communication and Network level*

Standardization bodies and industry associations (IETF, IEEE, Bluetooth SIG, Zigbee Alliance, etc.) have specified and developed protocols to enable energy efficient IoT by reducing communication overheads.

Also, researchers have focused on the optimization of access technologies, and create different connectivity options (BLE, Wi-Fi low power, NB-IoT, LTE-M, Zigbee, LoRa, etc.), on adaptation of IP protocols (6LoWPAN, RPL, and CoAP) to extend the web architecture to the most constrained sensors, and on developing lightweight protocols enabling the connection of almost everything to the cloud (MQTT, etc.). [24, 60, 9]. Each connectivity option offers different bandwidths, transmission ranges, energy costs, reliability level, etc. [68, 58, 5, 6]. Also, a number of studies such as [37], focuses specifically on the *access network* power consumption of a number of potential access network technologies and architectures which are modeled for a range of IoT traffic and background network traffic levels. It provides some insight into the energy efficiency of various access network technologies. [37]. Literature in energy optimization network level can be classified into four approaches *duty cycling (scheduling)* , *routing*, *data-driven* and *mobility* [13].

Duty cycling and Scheduling approaches [49, 41, 7, 91] focus on optimizing the networking subsystem by exploiting node redundancy by employing adaptive duty cycling. Early work in duty cycling [35, 42, 100, 67] assume that the sensor nodes are homogeneous and connected via homogeneous transmission methods that are energy constrained (battery, energy harvesting). Techniques such as Low-Energy Adaptive Clustering Hierarchy (LEACH)[42], in homogeneous networks, aims to distribute energy dissipation evenly so dubling the system lifetime.

In comparison, the IoT ecosystem is characterized by a larger level of heterogeneity. Clustering algorithms used in heterogeneous networks include techniques such as the Stable Election Protocol (SEP)[34], where nodes have two different energy levels, normal and advanced and the Energy efficient heterogeneous clustered (EEHC) technique [57] which assumes that a percentage of the population of static sensors are equipped with additional energy. Similarly, the Energy Efficient Clustering Scheme (EECS) [104], uses cluster heads, i.e. nodes with more residual energy as gathering points. Energy efficient cluster head election protocols[59] have been designed for heterogeneous networks. A distributed extension, Distributed Energy Efficient Clustering (DEEC) [80], assumes that all nodes in the sensor network are equipped with different levels of energy. In [103], they explore placing static nodes that have unlimited energy resources, so they have two levels, nodes that consume the most energy should be line powered and the leaf nodes that consume less power.

Routing approaches, such as [66, 76, 65, 106, 81] play an important role in the overall architecture of wireless sensor networks and the Internet of Things. As routing techniques propose energy efficient routing, forwarding, topology strategies to controls the transmission of the packets with in the IoT by selecting the optimal route to save energy and ensure a longer network lifetime. Routing protocols fall into three categories: *reactive*, which establishes a route on demand, *proactive*, which maintains the topology and updates the routing tables periodically, and *hybrid* protocols.

Data-driven approaches[96, 33, 21, 62, 44]focus on reducing data sampling and transmission by exploiting data aggregation, compression, or prediction. *Mobility* [13] is focused in having mobile entities, which can be the sink or the whole network nodes.

2.3 Optimize Energy for Computation at the *system level*

There has been several research efforts to modify IoT behavior based on the application requirements which can be classified into two approaches. *Quality driven* approaches [101, 77, 63, 54] that modify IoT behavior dynamically to conserve energy with meeting threshold such as QoS, in response to system changes. Secondly, *Event driven* approaches [99, 82] that achieve energy efficiency by selectively allowing the minimum IoT resources and triggers more IoT resources as needed based on the application demand.

Earlier literature in *system level* is based on wireless sensor networks which can be classified into three approaches *duty cycling*, *data-driven* and *mobility* [13]. *Duty cycling* approaches [35, 42, 100, 67] focus on optimizing the subsystem by exploiting active/sleep states and node redundancy. In early techniques, sensor nodes are homogeneous(limited heterogeneity[104]) and energy is constrained, connect via homogeneous transmission methods, e.g. LEACH[42] applied in homogeneous networks and aim to distribute energy dissipation evenly and doubling the system lifetime. *Data-driven* approaches [96, 33] focus on reducing data sampling and transmission by exploiting data aggregation, compression, or prediction. *Mobility* [13] is focused in mobile entities, which can be the sink.

Recent methods developed by researchers to tackle energy efficiency problem in IoT platforms are vary. There has been several research efforts to modify *IoT sensing* behavior based on the application requirements to conserve energy. The tradeoff decisions between energy costs and performance metrics [105, 95, 61], by designing an energy optimization sensor scheduling and activation algorithms with the consideration of quality-of-sensing and adjusting their sensing functions.

On the other hand, some IoT applications incur high communication and energy costs.

Managing IoT devices to reduce network overhead is critical and number of studies consider *transmission scheduling* to reduce energy consumption. They argue that devices consume more energy in communication. Their technique is a self-adaptive technique that aims to minimize the energy by harvesting in significant manner on IoT. Their approaches range from modifying scheduling strategy with irrelevant traffic streams support, optimizing graphs using critical path elimination, building mini clouds at relays, coordinator and gateway devices in an IoT network, reduced number of hops in the network, integrating routing techniques and node placement techniques in a single network architecture. [94, 8, 88].

Chapter 3

A Context-Aware Approach for Perpetual IoT Systems

In this chapter, we present our overall approach to understand and address realistic challenges in enabling perpetual monitoring deployments in community IoT systems. In particular, to gain realistic expectations of the issues in community perpetual IoT deployments and their potential solutions, we considered a popular mission-critical personal use-case, elderly fall detection, which was the personal sensing platform (SAFER) on our (SCALE Project) Safe Community Awareness and Alerting Network implementation [17, 1, 97]. Therefore, we designed and implemented SAFER, based on our real-world test-beds, we conducted measurement studies and identified major IoT perpetual monitoring challenges. A key challenge here is the energy consumption associated with perpetual operations and energy constrained associated with battery-powered devices. We argue that utilizing semantic, multiple heterogeneous sensors and real-time scheduling are required at cross levels to address trade-offs in energy cost, service quality, and latency. In this thesis, we develop end-to-end novel energy-aware optimization techniques across three layers: sensing, communication, and processing.

3.1 SAFER: Creation of a Multi-sensor Platform for Perpetual Services

The design and implementation of the SAFER multi-sensor fall detection platform [10, 11, 12] are among the contributions of SCALE the smart community project [1, 97, 17]. An important goal of SCALE research is to contribute to the quality of life of the elderly people and help them to maintain an independent lifestyle with the use of sensors, signal processing and the available communications infrastructure. From this perspective, SAFER was motivated and exploited by the SCALE as personal IoT awareness sensing system; it has been designed as a low-cost elderly fall detection system; as the detection of unusual human activities such as falling person is of utmost importance. An initial version of SAFER was deployed at a senior living facility in Montgomery County, MD.

A range of techniques and systems are undertaken by SCALE team, including SAFER [10, 11, 12], SCALECycle [107, 108], EnviroSCALE[83], geographically-correlated resilient overlay network (GeoCRON) [18, 19], resilient IoT data exchange (RIDE) [20], and AquaSCALE [40, 39]. In the following section, we briefly overview the SCALE system's architecture, design, and deployments.

3.1.1 SCALE: A IoT-based Community Awareness Platform

SCALE [17, 1, 97] project was developed as a response to the SmartAmerica challenge [2] by our team at University of California, Irvine (UCI), the government of Montgomery County, Maryland and several industrial partners, aiming to extend a smarter, safer home to everyone by leveraging the pervasive Internet of Things (IoT).

The goals of the SCALE project are[1]:

- Extend a connected safe home to everyone at a low incremental cost.
- Automatically detect emergency events, alert residents, confirm emergency via phone or app, and initiate contacting first responders
- Jump-start a live testbed for identifying and researching Internet of Things (IoT) challenges
- Connect disparate systems via an open multi-protocol data exchange
- Bring together key industry, academic, and government organizations to brainstorm, share ideas, and collaborate on prototype systems
- Expand community awareness and involvement in safety and IoT

SCALE [17] uses novel networking technologies, commodity sensor devices, cloud services, and middleware abstractions to sense, analyze, and act on sensed events in a distributed manner. SCALE has been successfully deployed in the Victory Senior Housing facility in Montgomery County, MD for a wide variety of sensing applications, such as personal safety, building/space safety, seismic activity, fire events and environmental monitoring (air quality). In addition, SCALE brought out a range of challenges that led to SCALE2, which is the second phase of SCALE with multiple deployments [97]. SCALE2 included a set of long-term research efforts that focused on improving the resilience/perpetually of IoT systems.

SCALE System's Architecture

The SCALE system architecture, Figure 3.1, shows four layers: devices, local network, cloud services, and applications. SCALE is an end-to-end IoT platform leverages sensors and communication platforms for a broad range of applications. It entails assorted access network technologies and their integration into a common platform for resilient/perpetual data collection/processing.

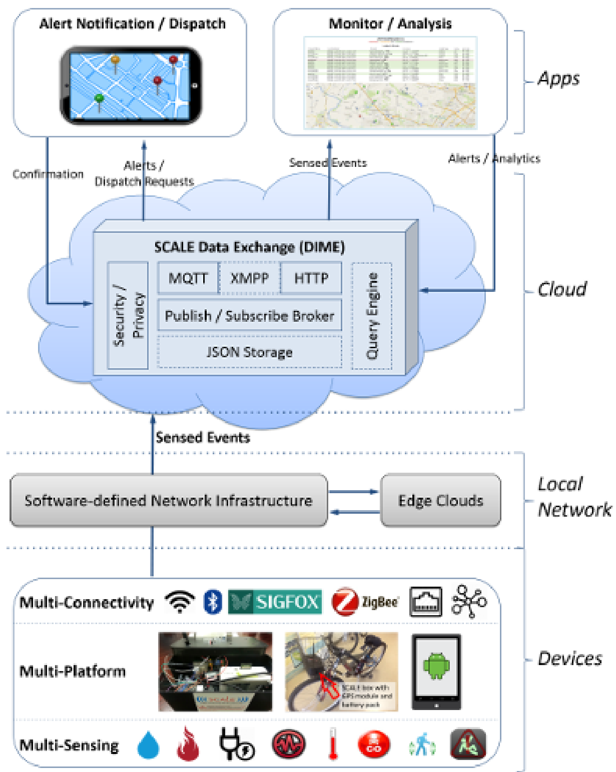
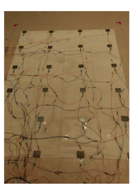


Figure 3.1: The SCALE system architecture, showing four layers: devices, local network, cloud services, and applications [1]

3.1.2 From SCALE to SAFER - Hardware and Software

The SAFER system architecture allows the participation of numerous types of devices, from low-profile embedded devices (e.g. Raspberry Pi) to general computing platforms (e.g. smartphones and PCs). Many devices are utilized to sense activities, each when used has created certain challenges. Therefore, we aim to study the combination of sensors to reduce the limitations and maximize the benefit of each type of sensing. Perpetual and resilient monitoring methods in this setting and combining multi sensors can enhance the performance under real-life conditions by improving the accuracy, reliability, usability, and power consumption. Having SAFER hardware platform allows us to further carry out tests and measurement with heterogeneous sensors, networks, etc. to support our perpetual research, validate our approach and explore challenges/insights that arise in real world deployments. In the following section, we overview the SAFER platform and activity recognition diverse sensors.

The testbed of SAFER, as shown in the following figure, Figure 3.2, includes many hardware(sensors)/software:



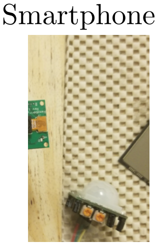
(a) Smart pad

(b) Ti Tag

(c) Smartphone Acc.

(d) Camera

(e) Acoustic sensor



(f) Motion sensor

Figure 3.2: SAFER IoT Devices

Mobile: Application and Accelerometer

The ubiquity of mobile smart devices provides users and developers significant capabilities due these devices' computing capabilities and sensors. Thus, we started our fall detection system by utilizing a smartphone-based fall detection method using the features of triaxial acceleration values of x, y, z which is obtained from the built-in accelerometer sensor embedded on Android smartphones. Many daily activities like sitting, walking, standing, lying, and running, plus anomalous events (e.g. falls) can be collected through accelerometer data. An Android application was built for personal fall detection called "Fall Detection"; Figure 3.3. It analyzes the device's accelerometer readings using the algorithm presented in [23] to distinguish between fall and non-fall. Upon detecting a user falling, the applications presents them with an option to cancel the alert, thus preventing false alarms, before a countdown timer expires and the phone publishes the alert via MQTT to call for help.

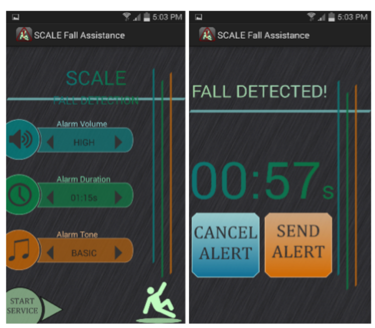


Figure 3.3: The Android Fall Detection Application

Wearable: Ti-Tag

We integrated a wearable sensor (Ti-tag CC 2541- CC2650) and mobile phone to increase the accuracy of capturing falls. Such as system develops an observing information from multiple sensors (phone accelerometer, wearable accelerometer, gyroscope, etc.) to detect a host of daily activities and anomalous events (e.g. falls). The Sensor Ti-tag CC 2541-CC2650, as shown in figure 3.4 runs on a battery and communicate via the Bluetooth Low

Energy (BLE) protocol with the Android device. Both have accelerometer, magnetometer, gyroscope, however CC 2541 has a microphone as well.

Figure 3.4: Ti SensorTag CC 2541- CC2650

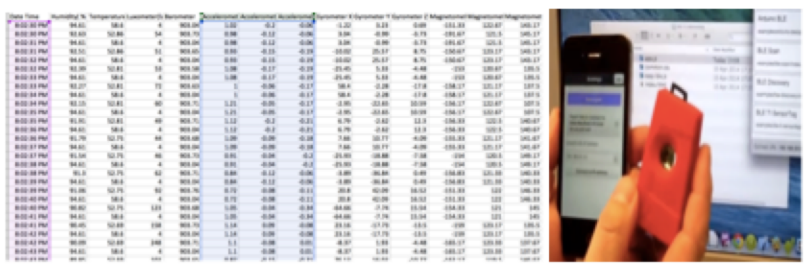


Figure 3.5: Ti-Tag Sensor Readings

Ambient: Smart Pad

We engineered a low-cost and privacy-friendly system ambient system [?]that observes the status of multi-connected sensors deployed on the ground to allow immediate assistance by detecting different activities of daily living (ADLs); then, contacting and sending alerts to healthcare contacts when a fall is detected. We implemented a 4 by 6 feet prototype using (FSR) pressure sensors in our DSM-UCI lab. We deployed the sensors on the ground on a balanced sensors distribution structure based on statistics of human body dimensions. Then, we gathered and processed the sensors' readings on-the-fly using a procedure that inspired by the Connected-components labeling algorithm and Union-find data structure. We considered inputs for five successive snapshot, 3.6, sensors samplings to detect a fall; once fall has been detected we send an alert message to the healthcare provider contact and publishe the alert via MQTT.

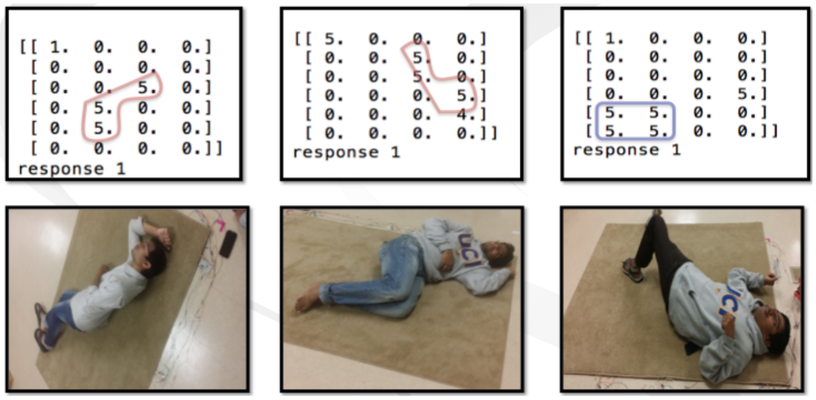


Figure 3.6: Smart Pad - Simulated falls in our implementation

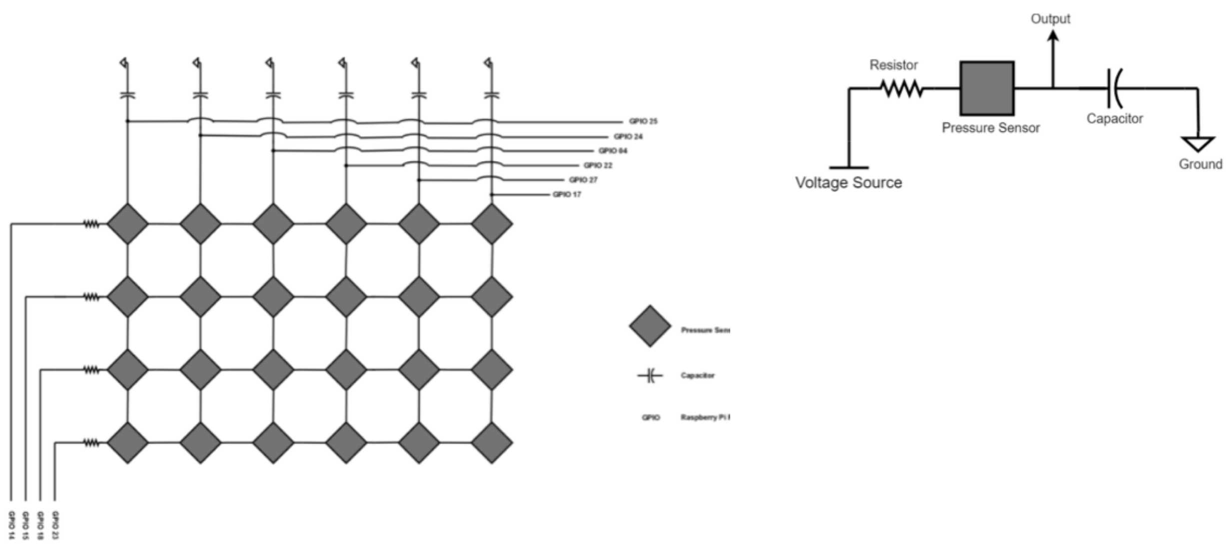


Figure 3.7: Smart Pad - 4*6 Matrix Architecture

Vision: Camera and Acoustic

An interesting direction that we considered next is to integrate a variety of multimodal sensors to confirm about emergency situations (falls in our setting). Audio signals provide important and essential pieces of evidence of the situation and complements the information from video signals in elderly care, home care, and home security. Screaming is one of the

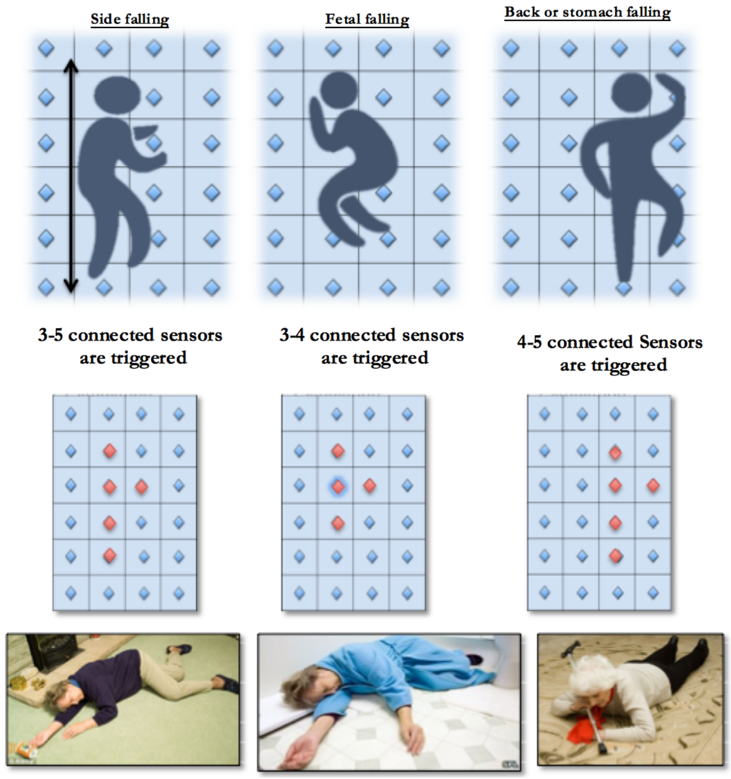


Figure 3.8: Smart Pad - Simulated falls based on the Connected-components labeling algorithm

events that is important in such environments for family members, caregivers, and security guard.

We built a tool that constantly records audio using a Raspberry-Pi USB microphone and when it detects a shout or a loud noise crosses a configurable threshold value (set to 300dB currently), it starts recording and the audio is extracted as a file. This recorded audio is sent to pre-trained machine learning model which classifies audio inputs as emergency situations such as a scream or a call for help. If the model identifies that as a human scream, it captures an image through a camera connected to the Raspberry Pi. This event based escalation approach is primarily done to minimize the false positives and to make sure that the privacy of users is not encroached upon. The captured image is published to SCALE, as shown in figure3.10, to confirm or reject the emergency and also for informing the emergency services.

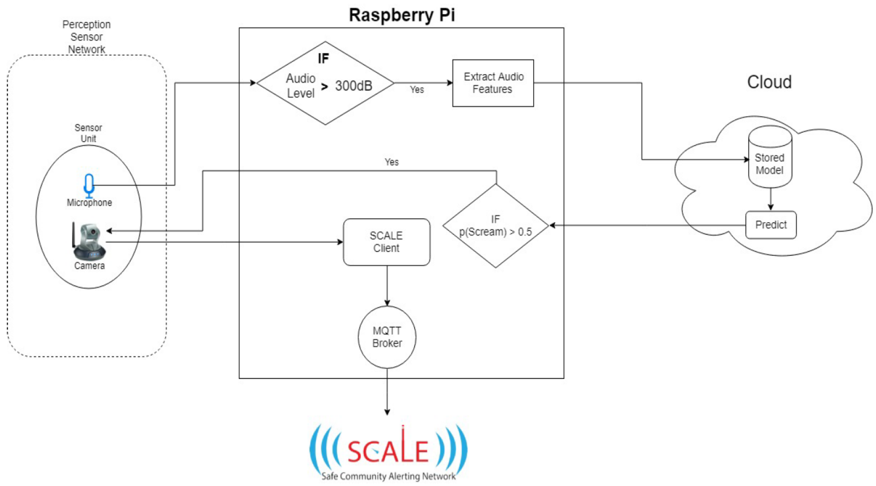


Figure 3.9: Architecture of the Vision System

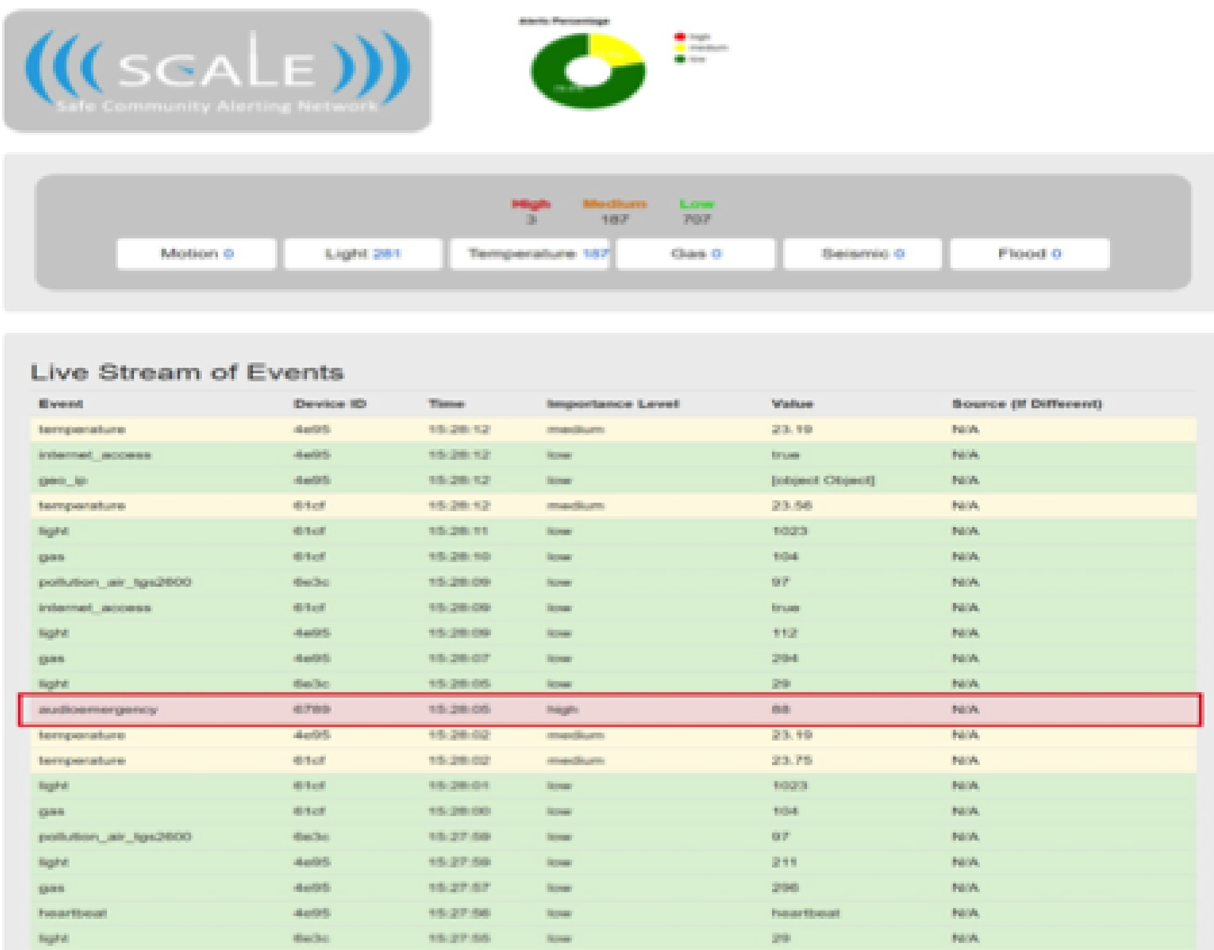


Figure 3.10: Data puplished on the back-end SCALE server using the MQTT protocol

Our experience in the design, implementation, operation, and maintenance of SAFER systems has demonstrated the key challenges of IoT-based services. While each approach in SAFER platform proves level of accuracy, they also suffer from several drawbacks. The ambient system suffer from reliability due to node failure, the wearable system suffered from power consumption, and cameras/microphones have privacy violations. We need to identify the challenges to exploit these different approaches and devices efficiently in our SAFER system, but the question is when and where, this can be answered by conduction measurement studies.

3.2 Lessons and Research Challenges

We had interesting observations and learned important lessons during the span of the projects through the maintenance work and measurement study, which helped us identify the end-to-end challenges and derive research problems; each of those challenges involve a large and complex scope [47].

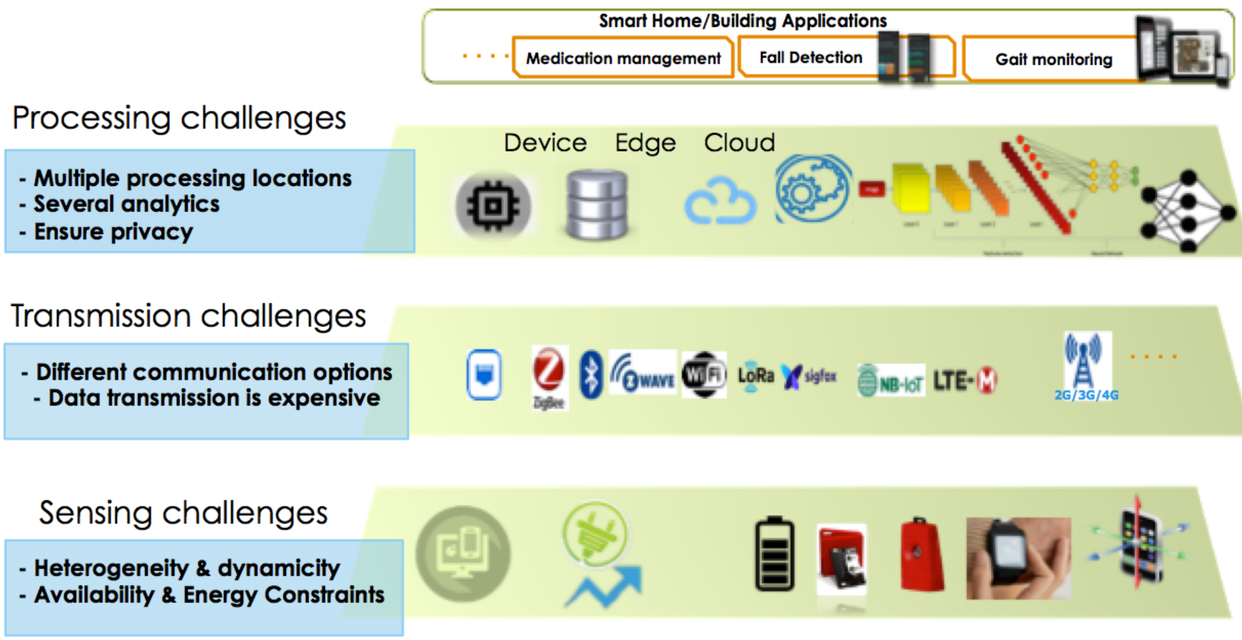


Figure 3.11: SAFER End-to-End Challenges

3.2.1 Energy Constraints

One of the crucial challenges mission-critical and assisted living settings is the energy limitations caused by perpetual operation and continuous energy consumption, which will be our focus. Execution lifetimes of IoT devices rely heavily on limited on-board battery capacity; this has an impact on service availability and in turn affects the quality of service delivered by these solutions. To the greatest extent IoT devices used in these settings are mobile and wearbles which are typically small in size with restricted resources including limited battery, need to be recharge or change frequently. Also, continues energy consumption of mobile devices will reduce the lifespan of the battery which can increase the cost of recycling lithium batteries and pollution caused by disposal batteries. In addition, each deployment includes some IoT wall-powered devices which rely on electricity; however, the cost of electricity consumed by wall-powered devices for sure someone should pay for it. Also, we need low-cost and mass-scale deployments in IoT systems to increase redundancy to overcome component variability and structural failures, and this raises the cost of energy.

3.2.2 Heterogeneity

Smart home deployments supporting a new way of living using data collected from different types of IoT devices. Subsequently, such diverse data are sent through non-Uniform connectivity that are analyzed, processed and utilized to enable multiple safety services. Various mission-critical and assisted living applications, such as fall detection, gait monitoring are needed. Thus, various requirements are needed to incorporate and facilitate efficient development of these systems. The heterogeneity in these settings comes across four levels: heterogeneity (in IoT devices, connectivity, processing, and applications).

IoT devices typically are small in size with restricted resources and divers capabilities including limited compute power, communication bandwidth, battery and storage/processing capabilities. SAFER uses various types of networking technologies in order to facilitate communications among heterogeneous sensor devices. Each sensor has diverse (often multiple) network technologies typically depends on the sensor application. Various wireless networking options exist, such as Wi-Fi, Bluetooth and ZigBee with different transmission cost, latency, etc. Wi-Fi is the most well-known wireless networking technology with a high power profile. Bluetooth, on the other hand, has one of the lowest power profiles and a limited range. 3G/4G/LET are good for outdoor deployment where WiFi access is limited, but they are costly. As these examples illustrate, the power needed and distance between sensors in a particular sensor application can dictate the configuration among each sensor configurations' options. Each application requires levels of accuracy/quality and latency to meet services demand.

3.2.3 Privacy

The personal nature of information that can be captured in instrumented physical spaces poses significant privacy risks. These privacy risks/issues prevent the full coverage of certain sensitive types of measurements in both public and private spaces. For example, in assisted living setting we cannot instrument cameras to monitor elderly fatal falls in one of the most hot spots at homes (e.g. bathrooms). Therefore, protecting individuals privacy is one of the main reasons for involving divers type of IoT devices (wearbles, ambient and vision) in those deployments, in order to confirm events in all spaces around the clock without violating privacy.

3.3 SAFER: Focused Energy Measurements Study

We conducted several iterations of energy measurement study using the SAFER platform we created. We collected energy cost and accuracy for multiple configurations for each approach in our testbed: smartphone, wearable, smart-pad, camera, etc. We started by measuring the energy cost of running a fall detection application on smartphone using the accelerometer data from different sources. We focus on testing the energy consumption when the system is actively running as a background service. Using a Monsoon Power Monitor, we measured two approaches for the fall application to receive accelerometer data.

The first test uses Android Nexus 4's built in the accelerometer sensor. Our testing utilizes a Nexus 4 device operating with Android 4.3 Jelly Bean, which is the first version in the Android platform that natively supports the Bluetooth Low Energy standard. The accelerometer used on the Nexus 4 is an Invensense MPU-6050 sensor, which functions as an accelerometer and gyroscope. The second test, use a CC 2541 Ti Tag sensor to send accelerometer data, via the Bluetooth Low Energy (BLE) protocol, to the Nexus 4's fall detection application. The Ti Sensor Tag uses KXT J9 Accelerometer, which only functions as an accelerometer. For each of these tests, we utilize a partial wake lock, which blocks the Android system from putting the CPU cores into sleep mode. If we allow the CPU to sleep, the on-chip accelerometer could provide inaccurate information or stop computing the fall detection algorithm.

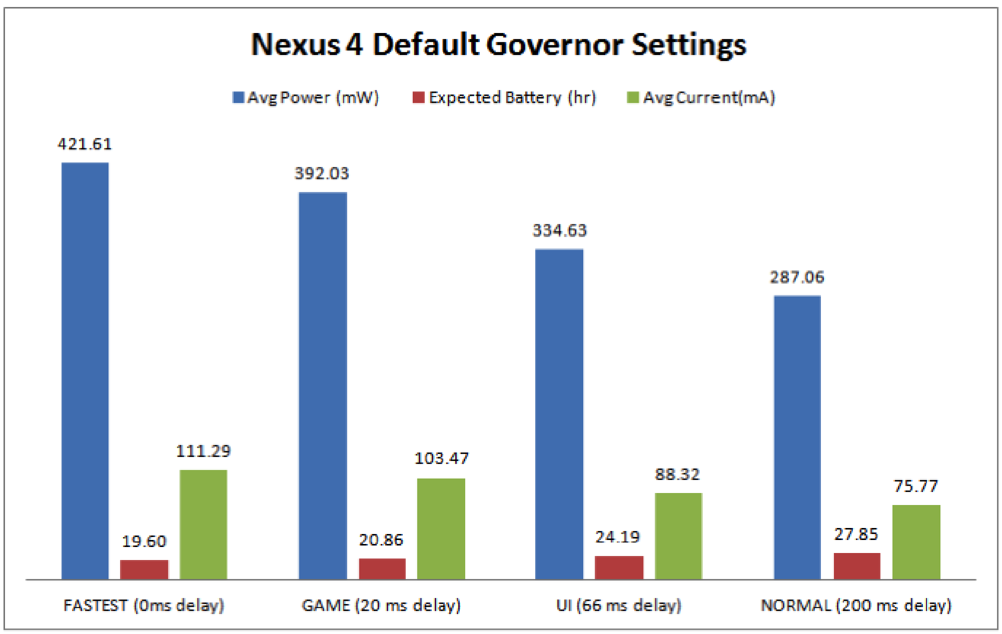


Figure 3.12: Measurements - Nexus 4 Running SAFER Fall Application

The Nexus 4 Android system provides four default delay updates, which are the following: FASTEST, GAME, NORMAL, and UI, as shown in Figure 3.12 . From the application level users can set a delayed rate, which affects the sensors ability to send information to the application layer. Each of these delay rates does not imply the Android will send data at that particular time, but rather a hint to the underlying device drive. This implies the hardware might provide either faster or slower data that the defined delay rate. From the Android source code, these four rates are defined the in the above figure. This difference can by the update frequency from the device. As a result of the high update frequency, it is clear that the faster update rate, will contribute to a significant consumption of energy. This is also aggravated with the usage the garbage collector and the data passing into the Java layer all the way from the accelerometer's device driver.

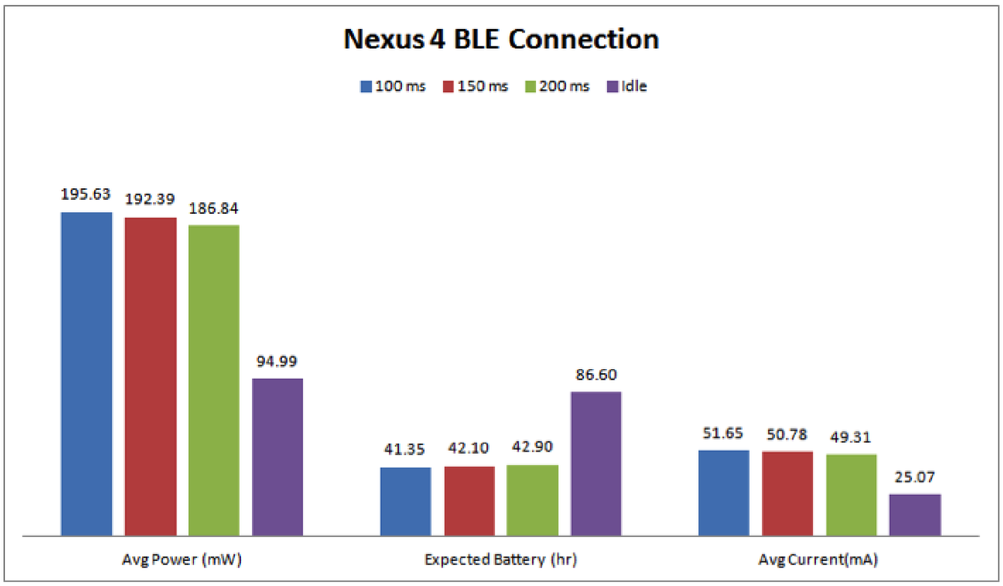


Figure 3.13: Measurements - Ti Sensor Tag Running SAFER Fall Application with BLE

Then, we measured the Ti Sensor Tag, which is a Bluetooth Low Energy dongle that provides its sensor data to the master device at fixed intervals. The fastest interval that the system allows is 100ms. The results, in Figure 3.13 show that the BLE approach reduces the energy consumption compared to the mobile on chip accelerometer. This could be due to the disabling of the accelerometer sensor or the reduced frequency in samples. Since when the on chip accelerometer is on, its normal operating current is 500 µA. On the Ti Sensor tag, the active power consumption is 135 µA. From our measurements we see the device as a whole saves 50% more power at 200 ms on the BLE methodology. This is primarily because the update frequency is lower than the on-chip methodology. Even though the accelerometer in the Sensor tag consumes less power, the Sensor Tag also runs on a battery, which means it's communication with the Android device is also limited by its constant Bluetooth transmission and sensor usage.

The results show that the BLE approach reduces the energy consumption, with the 50% improvement in power usage, compared to the on chip accelerometer. Even though the accelerometer in the Sensor tag consumes less power, the Sensor Tag also runs on a battery, which means it's communication with the Android device is also limited by its constant Bluetooth transmission and sensor usage.

In addition, we conducted accuracy and power consumption measurements for the smart pad, which is listed on Chapter4 Table 4.1.

Then, we investigated the energy consumption of various IoT access network technologies that exists in our testbed such as Wi-Fi, Bluetooth, NB-IoT, and LTE-M, to present their average energy consumption, which has been shown on Chapter5 Table 5.2.

Through SAFER platform and measurements, we recognized and identified opportunities for improved operations that will be illustrated in the following solution strategy section.

3.4 SAFER Solution Strategy: Cross-Layer context-aware Optimization for perpetual IoT monitoring

In this thesis, we primarily focus on the personal sensing deployment to experiment with various personal and community safety-oriented applications and its challenges. During the years of SAFER systems designing, implementation, operation and maintenance, we collected measurements and gained experience/insights that motivated the need for exploiting the space semantic, activities, and devices/networking/processing heterogeneity through intelligent scheduling/activation to optimize system lifetime and reduce perpetual energy cost.

Our SAFER framework run across-layers, taking into consideration the context of the space; plus the profiles/states of the available IoT devices, network technologies and processing locations to meet the overall application accuracy/quality and latency.

3.4.1 Design Intuitions

In this section, we overview the design intuitions that were leveraged by SAFER system's architecture and design.

Exploit Space segmentation

Partitioning is a viable strategy of overall system efficiency, scalability and performance[22]. Specifically, we compute a floor-plan segmentation of the space being instrumented and monitored. It is desirable step if the application only needs IoT devices in designate areas in the floor plan; i.e. not all IoT devices are utilized all the time to monitor falls for independent-living residents.

Understand IoT Devices' Profiles

Knowledge of device capabilities can be also used to activate an adequate subset of IoT devices to meet the demanded quality of service. Applications are part of smart building/home systems that deliver multiple services. Each instrumented IoT device comprises one to many sensors (data sources) that feed data to a designated service. Sensors can be of type video, audio, motion, 3D accelerometer, gas, temperature, and others. Also, each IoT mostly has one to many network interfaces (interfaces) to communicate and send their data. Interfaces may be WiFi, Bluetooth, NB-IoT, LTE-M, Ethernet, Zigbee and others.

Thus, each IoT device can operate in different configurations in terms of choosing values for their different operating parameters, such as communication intervals, sampling rate, computation frequency and a subset of sensors and one interface at any given time. While the choice of sensors depends on the service for which the node is sending to, the choice of interface depends on many aspects because the variation of these technologies results in different energy consumption rates and varying degrees of latency due to varying data transmission rate (bandwidth) associated with each network interface.

Leverage Semantics: ADLs and activity patterns

Knowledge of the activities of daily living (ADLs) of a resident can provide us with information about the location and activity type; this can be utilized intelligently to reserve resources (e.g. minimize energy dissipation) in the integrated system. While the crucial event probability fluctuates during different activity of daily living (ADLs). For example, more than 70% of the falls happen in the rooms, when patients move to the bed. Therefore, aggregating multiple low cost sensors as a base-line ambient sensing, (motion, door sensors, temperature, etc.) To detect events and activities. These base-line sensors can be wired or wireless with a slow-battery-drain and consume power at a very low rate (e.g. less than 8 watts consumption in layout of the 39 sensors (31 motion, 4 door, and 4 temperature sensors) deployed on the 900 ft2 home).

Prioritize Real-time events

Thus, recognizing ambient events from these low cost sensors and context information, such as ADLs, occupancy information is beneficial to reduce wasteful resources. This illustrates the need to provide abstractions that capture the dynamicity of the underlying space into multiple modes that can then be used to trigger increased sensing based on some shifting events. We investigated the abstractions for exposing indoor space dynamicity in IoT deployments to build more reliable and efficient systems. We argue that in the indoor setting, there are multiple modes that the physical space shifts between; this is based on the occurrence of events, as shown in Figure 3.14, We refer to these settings as space-states.

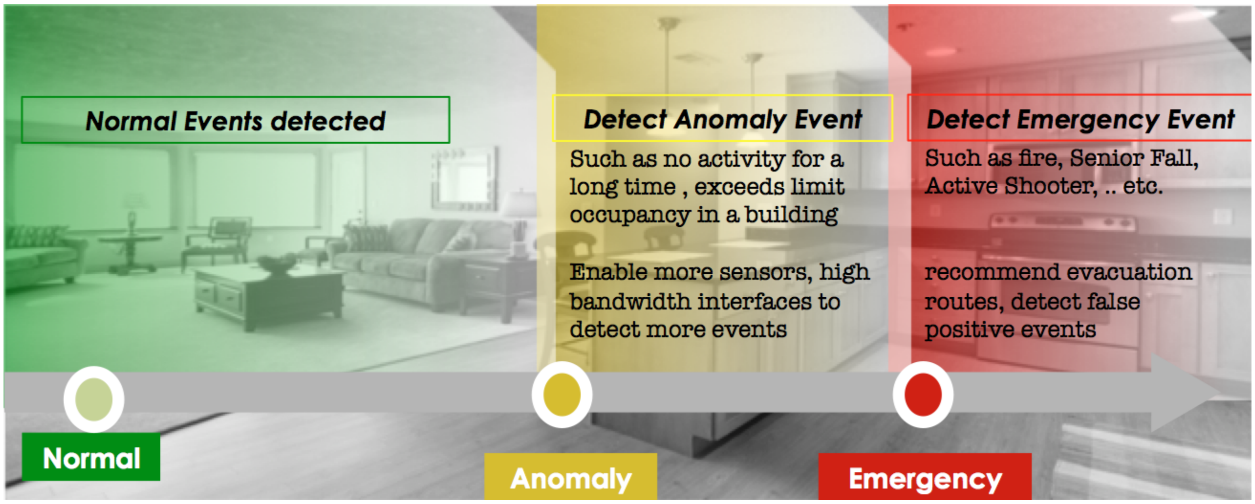


Figure 3.14: Multi space-sates based on events occurring

We defined three modes (Normal, Anomaly, and Emergency) under different conditions based on the amount/quality of data that needs to be sensed and transmitted. For instance, in the safety system after a fire event (smoke, heat, gas detection), the space-state shifts to the emergency state where we have low latency and high data quality - we enable more sensors, high bandwidth interfaces to detect the causes of fire, and recommend evacuation routes. On the other hand, the system will switch to the anomaly space-state if there is a crowd (high occupancy) in a closed area, which requires more data to identify the causes and actions; a latency tradeoff occurs to capture more information in the anomaly mode.

In shared spaces, we define these space states to be personal space-state, as every individual is surrounded by unique space state based on the recognized activity and personal profile, this illustrated in more details in chapter 66.

Seeking to address the major energy challenge that emerges from exploiting IoT deployment in perpetual mission-critical applications, we leverage the above intuitions and knowledge of context with scheduling techniques to meet application needs. We propose cross-layered context-aware architecture for perpetual monitoring in safe communities; SAFER is an approach that utilized semantics to build an efficient end-to-end IoT systems from sensing to processing. This architecture provides for the efficient inclusion of functionalities that range from pre-data-related collection to real-time context knowledge extraction. Specifically, we make intelligent activation in three levels: sensing, communication and processing. This section provides a brief overview of our approaches; technical details are expanded in the following chapters.

3.4.2 System Architecture and Middleware Design

Developing an optimal resource efficient system for perpetual and heterogeneous IoT operation needs comprehensive knowledge about the floor-plan architecture, individual's activity patterns and IoT device status. To handle this complexity that arises due to dynamic nature and diversity of the underlying ADLs and IoT devices, we propose a phased system framework (Figure 3.15).

Figure 3.15 shows in the lower level heterogeneous IoT devices (wearable, ambient, and vision) are scattered in a smart home; these IoT devices have varying capabilities (power source, battery lifetime, connectivity, reliability and accuracy) that can be used as part of any application or critical event detection task.

Apriori Data collection

In first, we aim to capture the deployment setting. Specifically, we compute a floor-plan segmentation of the space being instrumented and monitored, the IoT device profiles, in-

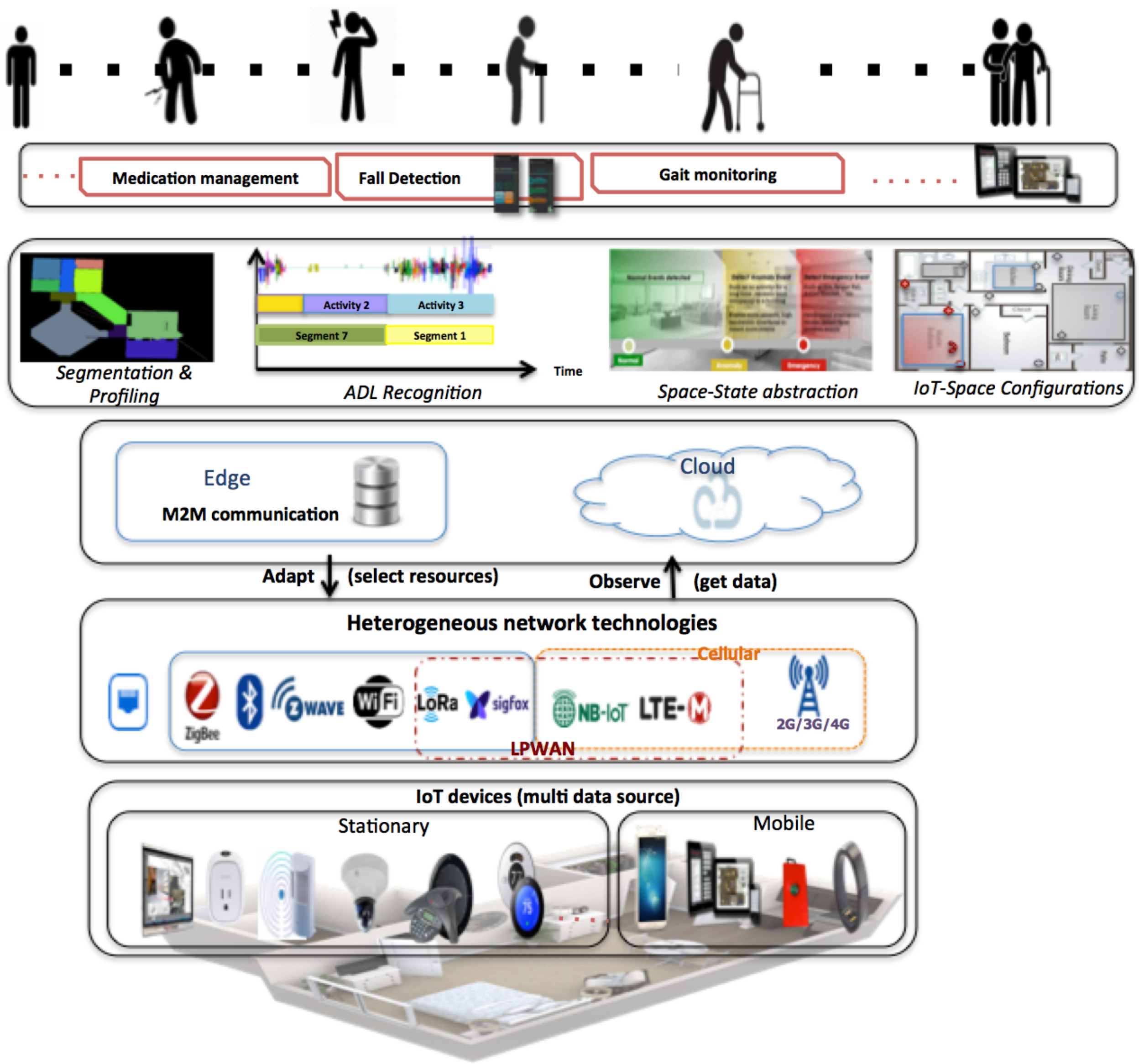


Figure 3.15: System Architecture

cluding status and different configurations given infrastructure information; we also leverage an elementary activity pattern and individuals' profiles that are provided by the user.

Real-time context knowledge extraction

In the second phase, we utilized the infrastructure knowledge to develop a sliding window based approach to perform activity recognition in a streaming fashion; recognizing activities and when new occupancy (ambient) sensor events are recorded, such as motion sensors.

Based on the previous phases and application needs, the status of participating IoT devices, sensing data rates, network interfaces, processing locations are adjusted at run-time. Note that this context-aware optimization approach can be applied to any heterogeneous IoT deployment setting (independent of layout and instrumentation) and can be configured to preserve the desired Quality of Service (QoS).

3.5 Chapter Discussion and Focused Techniques in Future Chapters

This leads to our three stages of optimization in the IoT system architecture (device, communication, and processing) and a semantic approach that is presented in details in Chapter 4, Chapter 5 and Chapter 6. In Chapter 4, we tackle the challenge of IoT devices heterogeneity (wearable, ambient, and vision) in terms of: energy cost, energy source (battery-operated and wall-powered IoT devices), processing capability, mobility by utilizing the context of extracted activities of daily living (ADLs), time and location for energy-optimized sensor activations.

Then in Chapter 5, we considered how best to exploit the dynamic space-states and knowledge of the application needs in the underlying space with the challenge of the presence of multiple sensing modalities and multiple communication technologies and transmission protocols (such as: NB-IoT, LTE-M, LoRa, Wi-Fi, 4G/5G, Bluetooth, Zigbee, etc.) to intelligently activate the underlying systems' sensing and communication configurations. After this in Chapter 6, we discuss the real-time computation efficiency challenge to analyze insights obtained from sensory data along with scaling-up our setting with multiple individuals. We expanded our semantics to include a personalized space-states transition approach.

Chapter 4

Exploiting Semantics for Energy-Efficient Sensing

In this chapter, we explore techniques for optimization of perpetual IoT systems' at the sensing layer. We develop methods to support effective generation/sensing of data from heterogeneous devices in an assisted living context, specifically, an independent assisted living context. In this stage, our goal is to efficiently meet application requirements- data accuracy,leading to safety of occupants, while handling the energy limitations caused by perpetual sensing and collection of data from heterogeneous IoT devices.

4.1 Chapter Overview

As discussed earlier, perpetual operation of IoT devices is a challenge due to energy limitations. Execution lifetimes of IoT devices rely heavily on limited on-board battery capacity; this has an impact on service availability and in turn affects the quality of service delivered by these solutions. Our key idea is to exploit heterogeneity of IoT devices and semantics of events, i.e. knowledge of the activities of daily living to create an energy-efficient perpetual IoT system without loss of service quality. Using a multistage approach that models the behavior of humans in a space, we develop intelligent device activation techniques that can enhance the effectiveness of the IoT awareness system, both in terms of energy consumption and accuracy.

IoT deployments are heterogeneous - it includes devices with varying capabilities in terms of computation, sensing, energy source, energy consumption, mobility and communication; these devices use diverse communication protocols and direct connections to cloud platforms. Perpetual IoT applications incur high communication and energy costs; managing the number of active devices to reduce network overhead is critical. We aim to design a system model that alters the state of IoT devices and communication by utilizing real time semantic knowledge on user activities. We exploit the heterogeneity of IoT devices, multiple communication networks, current environment conditions and the *activity patterns* being inferred to enable energy efficiency. To lend focus, we develop our techniques in a target application domain - our IoT-based assisted living system aims to provide occupant safety while ensuring energy efficiency. Note that accuracy of event detection is a primary goal - missing a critical event (i.e. injury or fall) as a result of energy optimizations is unacceptable.

Outcomes:

- Formalizing energy efficiency of perpetual Iot-based awareness systems as a constrained optimization problem, that aims to optimize system lifetime and overall energy cost, which we show to be NP-hard.

- Designing a framework and associated algorithms for smart homes, our proposed approach operates in three phases combines a floor-plan segmentation algorithm[22], an activity recognition technique[56], and energy optimization algorithms with heterogeneous IoT devices.

- Developing and validating a prototype heterogeneous IoT system,"SAFER: an elderly fall detection IoT system" in a real world testbed.
- Conducting extensive evaluations to study the scalability and effectiveness of our algorithms and approach using simulation studies.

4.2 Approach Overview

Consider a home in an assisted-care setting with a single resident, John, that experiences frequent falls, where heterogeneous IoT devices (wearable, ambient, and vision) are distributed around the home. These IoT devices have varying capabilities (power source, battery lifetime, connectivity, reliability and accuracy) and are used as a part of an elderly fall detection system (SAFER). In general, they can be used for any critical event detection task. Battery-powered devices such as mobile and wearable sensors, worn by John dissipate power quickly and need to be recharged. Through the elderly fall detection system, we recognized and identified problems and opportunities for improved operations.

To capture John's activities in his home, we begin by designating areas in the floor plan that are used in predictable ways. In addition, knowing John's activities can provide us with information about the location and type of activity; this can be utilized intelligently to minimize energy dissipation in the integrated system. Knowledge of device capabilities can be also used to activate an adequate subset of IoT devices to meet the accuracy demand levels.

Our goal is to minimize energy consumption of the integrated IoT deployment (battery and wall-powered) to enable long-term operation while meeting accuracy threshold demands. We formalize the energy efficiency problem for heterogeneous IoT devices as a constrained optimization problem (proven to be NP-hard). Note that this optimization problem can

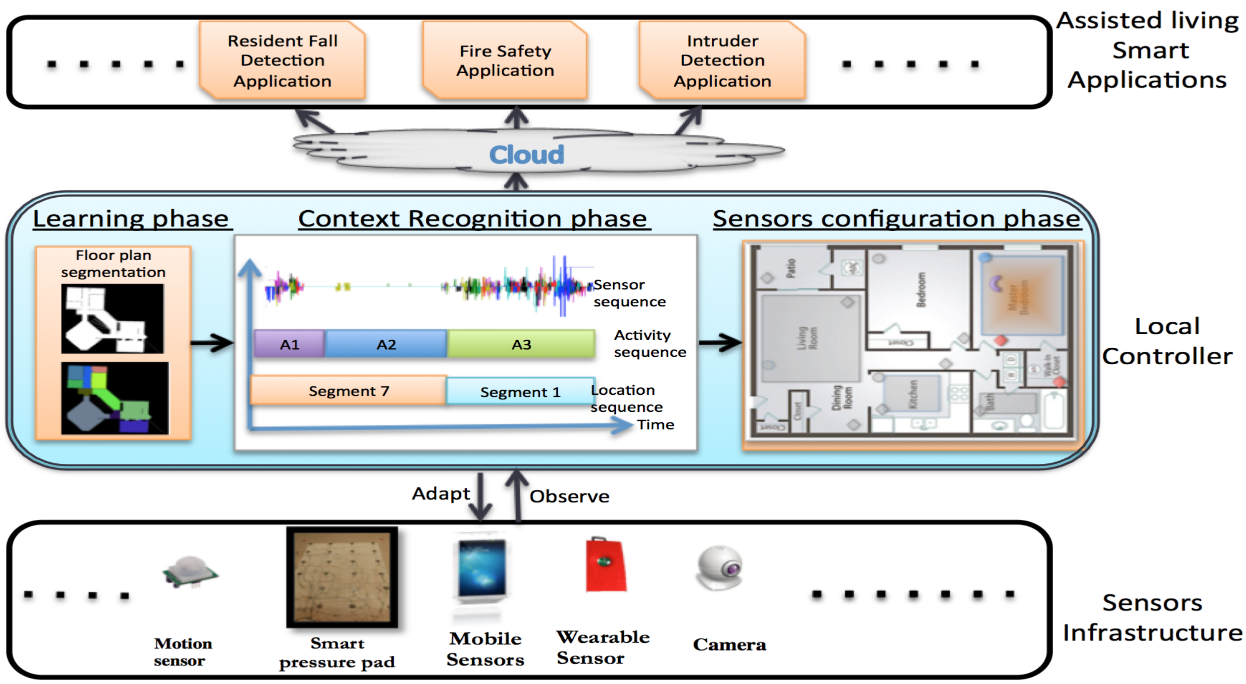


Figure 4.1: The SAFER Architecture illustrating the Three Phase Approach

be applied to any heterogeneous IoT deployment setting (independent of layout and instrumentation) and can be configured to preserve the desired accuracy thresholds. To handle complexity that arises due to dynamic nature and diversity of the underlying ADLs and IoT devices, we propose a three-phase system framework (Figure 4.1). In the first phase, i.e. the *learning phase*, we aim to capture the deployment setting. Specifically, we compute a floor-plan segmentation [22] of the space being instrumented and monitored, the IoT device profiles, including status and different configurations given infrastructure information; we also leverage an elementary activity pattern that is provided by the user. In the second *context recognition phase*, we utilize the infrastructure knowledge to develop a sliding window based approach [56] to perform activity recognition in a streaming fashion; recognizing activities as and when new occupancy sensor events are recorded, such as motion sensors. In the third phase, *the configuration phase*, the status of participating IoT devices is adjusted at run-time based on the current activity location and type. We design a dynamic configuration algorithm that is executed on the local controller to control the IoT network and to compute and realize the optimal overall energy configuration.

4.3 The Device Energy Optimization Problem for Perpetual IoT Systems

In this section, we discuss our assumptions and define frequently used terms and notations. With the assumptions, we formulate the heterogeneous IoT devices energy problem as optimization problem that we show to be NP hard.

4.3.1 Assumptions

Our approach targets independent assisted living systems; the dataset and recognition techniques are completely applied for one resident [28]. Moreover, we assume that in each home the presence of a *local controller*, which is a fixed device that has the ability to connect with all existing IoT devices via different access networks and protocols, such as Wi-Fi, Bluetooth, Ethernet, and ZigBee, etc. to track and manage the IoT devices' status. Also, we assumed that all IoT devices are operating independently in terms of detecting the critical event they are monitoring.

4.3.2 Terms and Notations

To capture activity context, we utilize a well-known abstraction i.e *Activities of daily living (ADLs)*, which are routine activities that people tend to do every day without needing assistance, such as cooking, sleeping, eating, moving, etc. Additionally, we define a *critical event* to be an incident that has a high consequence to the health of the individual, such as fall of an elderly person in the home. All IoT devices deployed in the scene are intended to monitor the person and detect the critical event should it occur. The floor-plan is divided into several *segments*, where each segment is a semantic sub area of the home, such as the living room, that consists of a set of IoT devices. We classify IoT devices in two groups in terms of their source of energy/power: unconstrained *wall-powered devices* that are connected to the energy grid all the time; constrained *battery-powered devices* that rely on their own limited battery or energy harvesting throughout their lifetime. Each IoT device can operate in different *configurations* in terms of choosing values for their different operating parameters, such as communication intervals, sampling rate and computation frequency. The variation of these values results in different amount of energy consumption rate and varying degree of accuracy level across the different configurations of a particular IoT device. We define *accuracy* of an IoT device for a given configuration to be the probability of detecting the critical event when the device operates in that configuration for a certain amount of time (referred to as the operation cycle). Obviously, there is a trade-off between the energy consumed by a device at its different configurations and the corresponding accuracy levels. Higher accuracy is desired but this comes at the cost of higher energy consumption, which leads to shorter system lifetimes (due to power drain on the battery-powered devices). The goal is to choose configurations for the devices to enable better system lifetimes.

4.3.3 Problem Formulation

We formulate our key problem of ENERGY OPTIMIZATION FOR HETEROGENEOUS IOT DEVICES as a constrained optimization problem. Assume a set of n heterogeneous IoT devices in a certain segment, $i = 1, \cdots, n$. Each device can be described by a *profile*, which consists of different configurations in which the device can operate. Let device i have l_i configurations, and e_{ik} and a_{ik} denote the rate of energy consumption and accuracy level respectively for configuration k of device i ($1 \leq k \leq l_i$). In any time instant, each device i has a residual battery capacity, denoted by r_i, at a certain time. Note that for wall-powered devices r_i is not defined or assumed to be ∞. We assume devices operate in cycles/epochs, i.e. once a configuration is chosen for a device, it operates for a certain amount of time before the next configuration is chosen. This duration, the operation cycle, is denoted by T.

The context includes set of predefined ADLs for the independent living user in a space, each ADLs has its own *demand accuracy*, denoted by τ. The demand accuracy is the level of accuracy of data that all *active* devices should produce in order to effectively capture the ADL.

Second phase should recognize the performed activity with a level of uncertainty which is a challenge .To overcome this challenge we obtain the formulation as follow by considering the combined demanded accuracy which is the probability that at least one among the set of the expected recognized activities is performing by the monitored patient.

We argue that when multiple devices monitor the critical event, the combined accuracy increases. For example, if two devices independently detect the critical event with adequate accuracy, i.e. if the probability of detecting the critical event, equal to a_1 and a_2, then the combined accuracy will be the probability that at least one of them is detecting the event. That is:

$$\text{combined accuracy} = 1 - (1 - a_1)(1 - a_2)$$

The demand accuracy is a variable that changes depending on the daily activity performed. For example, the demand accuracy will be higher if the individual is cooking versus if he is sleeping. The more the activity is crucial, the higher the demand accuracy should be. Therefore, the demanded accuracy τ for each ADL can be defined in many ways. It can be defined based on labeled activity pre-training phase model. Another way is to increase the demand accuracy when the intensity of occupancy sensors' readings are increased. In addition, it can be prescribed by the supervising physician which we considered in this paper. We want to select the optimal IoT devices subset with their appropriate configurations, so that the total energy expenditure remains as low as possible, while keeping the expected level of accuracy from the selected configurations above the demanded activity's accuracy level τ.

This subset selection minimizes the overall energy consumption and maximizes the battery-powered devices lifetime. In order to minimize the overall energy consumption, we should consider the device's energy consumption rate at its different configurations and the number of active devices (depending on the demand accuracy). We also need to extend the lifetime of battery-powered devices, so we should consider the remaining battery capacity of those devices. Considering all these issues, we define a *cost function*, denoted by c_{ik}, for each configuration of an IoT device as follows:

$$c_{ik} = \eta_i \cdot e_{ik} \cdot T \tag{4.1}$$

The cost function captures the "cost" of operating device i in configuration k. The cost is directly proportional to the amount of energy consumed during the cycle, which is $e_{ik}.T$. The cost also takes into account the fact that operating a battery-operated device is costlier than an equivalent wall-powered device when they both consume the same amount of energy. This is because battery-powered device runs on battery and their life depends on the remaining battery capacity. The operation arguably gets costlier when the remaining battery capacity becomes low. To reflect this, we multiply the base energy consumption with an adjustment factor, η_i, which is given by:

$$\eta_i = 1 + \beta \cdot \exp\left(-\frac{r_i}{r_i^0}\right) \tag{4.2}$$

where r_i^0 denotes the initial battery capacity of the device and β is a tunable parameter to adjust the effect. Obviously, for wall-powered device, we have $r_i^0 = \infty$, hence $\eta_i = 1$. For battery-operated devices, $\eta_i > 1$. Particularly, at the beginning (when r_i and r_i^0 are equal), $\eta_i = 1 + \beta.\frac{1}{e}$, and then η_i progressively takes higher value (as the r_i declines) until it reaches to $1 + \beta$ when there is no battery power left (i.e., $r_i = 0$).

We obtain an optimization formulation that chooses the configurations minimizing the overall cost of operation subject to the constraint that the combined accuracy level remains equal or above the demand accuracy. For the ease of exposition, we introduce an *idle* configuration (configuration 1) for each device that has zero accuracy at zero or low cost. This allows all devices to be operating exactly one configuration. We denote x_{ik} to be the binary variable that indicates whether we choose configuration k of device i. Given an i.i.d. assumption i.e. all IoT devices are independent, the combined probability that all IoT devices detect the critical event can be written as follows.

$$1 - \prod_{i=1}^{n} \prod_{k=1}^{l_i} (1 - x_{ik} \cdot a_{ik}) \geq \tau \tag{4.3}$$

Accordingly, we have the following optimization problem:

$$\begin{aligned}
\text{minimize} \quad & \sum_{i=1}^{n} \sum_{k=1}^{l_i} x_{ik} \cdot c_{ik} && (4.4)\\
\text{subject to} \quad & 1 - \prod_{i=1}^{n} \prod_{k=1}^{l_i} (1 - x_{ik} \cdot a_{ik}) \geq \tau && (4.5)\\
& \sum_{k=1}^{l_i} x_{ik} = 1, \forall i && (4.6)\\
& \forall x_{ik} \in \{0, 1\}, \forall i = 1, \cdots n, \forall k = 1, \cdots l_i
\end{aligned}$$

We can simplify constraint (6.2) as follows:

$$\begin{aligned}
& 1 - \prod_{i=1}^{n} \prod_{k=1}^{l_i} (1 - x_{ik} \cdot a_{ik}) \geq \tau \\
& \ln \prod_{i=1}^{n} \prod_{k=1}^{l_i} (1 - x_{ik} \cdot a_{ik}) \leq \ln(1 - \tau) \\
& \sum_{i=1}^{n} \sum_{k=1}^{l_i} x_{ik} \cdot \ln(1 - a_{ik}) \leq \ln(1 - \tau) && (4.7)
\end{aligned}$$

Consequently, we obtain:

$$
\begin{aligned}
\text{minimize} \qquad & \sum_{i=1}^{n}\sum_{k=1}^{l_i} x_{ik} \cdot c_{ik} \\
\text{subject to} \qquad & \sum_{i=1}^{n}\sum_{k=1}^{l_i} x_{ik} \cdot \ln(1 - a_{ik}) \leq \ln(1 - \tau) \\
& \sum_{k=1}^{l_i} x_{ik} = 1, \forall i
\end{aligned} \tag{4.8}
$$

The ENERGY OPTIMIZATION FOR HETEROGENEOUS IoT DEVICES PROBLEM is an NP-hard problem that can be reduced from the Minimum Multiple Choice Knapsack Problem. The knapsack problem is known to be a well-studied NP-hard problem and a special case of the multiple choice knapsack problem with the feature that each item is in a group of its own [53].

In the MINIMUM MULTIPLE CHOICE KNAPSACK PROBLEM there is a set of items which are partitioned into groups and each item has a benefit and a weight. The objective of the MMKP is to find the least profitable set of items such that the total weight of the selected items is at least the weight limit [48].

Similarly, in the HETEROGENEOUS IoT DEVICES ENERGY OPTIMIZATION PROBLEM, the goal is to select a set of IoT devices that minimize the total cost with exceeding the activity's accuracy threshold. Each IoT device has a set of configurations, including the option of not selecting it. Therefore, each IoT device defines a class from which we are selecting at most one option.

4.4 Algorithms and Heuristics for Device Energy Optimization in Perpetual IoT Systems

One naive approach we can consider is to *activate all the IoT devices* all the time. Obviously, that would deplete energy from all devices without a significant impact on system accuracy. In this section, we propose a set of feasible techniques to energy efficient perpetual IoT operation.

4.4.1 Greedy Algorithms

1) Balanced Remaining Battery Lifetime (BRBL)

In this approach, we activate the wall-powered devices first and then choose the battery-operated devices in descending order of their remaining battery capacity until we exceed the current activity's accuracy threshold. This algorithm inspired from some related work which schedule sensors activation based on its residual energy, such as [31]

2) Cost Function Gradient (CFG)

This solution is the greedy heuristic solution to our formulated MCKP problem. It is known that linear relaxation of MCKP [90], in which the indicator variables x_{ik} can be assigned real values instead of binary $0, 1$ values, that can be optimally solved by a greedy algorithm. As we cannot add a fractional IoT device, we will use the greedy integral algorithm with multiple configurations selections for each device as fractional placement. The algorithm starts with an empty set with zero overall benefit (delivered accuracy) and zero cost (energy consumption). It then makes a sequence of changes in which the selected IoT devices' configurations are upgraded to a more accurate option with higher overall benefits. The process continues until the demanded accuracy is achieved.

The pseudo-code for CFG is given in the following algorithm.

Algorithm 1: IoTSelectionBasedonCFG(τ, s)

```
Initialize CombinedAccuracy = 0;
Initialize list(i,k)  all the available IoT devices with its configurations k in this
  segment;
Enable selectedIoT(i,k)  the lowest configuration k for all available IoT devices;
forall list(i,k) do
    cost(i,k)= e_ik · (1 + β · exp(−r_i / r_i^0))
end
foreach list(i,k) do
    deltaA = accuracy(i, k_next) − accuracy(i, k_cur)
    deltaC = cost(i, k_next) − cost(i, k_cur)
    Calculate slope(i,k) = deltaA/deltaC
end
while (CombinedAccuracy ≤ τ) do
    Select i_k with the largest slope;
    selectedIoT(i,k) =1 ;
    forall selectedIoT(i,k) do
        PiAccuracy∗ = (1 − accuracy(i,k));
    end
    CombinedAccuracy = (1 − PiAccuracy);
end
```

4.4.2 The Dynamic priority scheduling Algorithms

Dynamic priority scheduling is a scheduling algorithm in which the priorities are calculated during the execution of the system. The goal of this on-line scheduling is to adapt IoT devices' configuration dynamically to ensure that energy resources are utilized for higher profiling events. The priority policy can be based on the location of the monitored user. In this approach, we activate the IoT devices that are present in the segment area where the user is currently in.

Furthermore, the second approach is to activate a subset of IoT devices based on power supply policy. Since our goal is to keep the battery-powered devices alive for longer time, we implement a second approach that gives priority to the wall-powered devices and activates them first until the combined accuracy exceeds the demand accuracy for the current activity.

4.5 Initial Measurements on our Real Testbeds

The testbed that we implemented to evaluate our technique, as shown in Figure 4.2, includes (a) a smart pressure pad which is a 4x6 matrix consisting of 24 Square Force-Sensitive Resistor sensors; (b) wearable sensor (CC2541 Ti SensorTag); (c) mobile sensor (accelerometer); (d) a camera (INSTEON). A local broker (Raspberry Pi B) supports the interconnections of multiple networks and sensors, and publishes data to the back-end SCALE server using the MQTT protocol.

(a) Smart pad (b) Ti Tag (c) Accelerometer

Figure 4.2: SAFER Prototype's IoT Devices

Then, we measured the energy consumption and accuracy of fall detection for each participating IoT device in different configurations. The values are shown in Table 4.1.

For the mobile phone experiments, we used an Android Nexus 4 mobile with BLE support and an Invensense MPU-6050 accelerometer, which functions as an accelerometer and a gyroscope. The Nexus 4 system provides four default sampling rates: Fastest (5ms), Game (20ms), UI (66ms), and Normal (200ms). The sampling rate affects the sensor's ability to send information to the application layer.

A higher sampling rate delivers more frequent updates leading to more accurate results; however, this also consumes higher values of energy. The wearable device we used in the

IoT Device	**Battery capacity**	**Configurations**	**Power consumption**	**Performance accuracy**
Mobile phone	**8**Wh	Fastest	421 mW	70%
		Game	392 mW	65%
		UI	334 mW	50%
		Normal	287 mW	40%
		Idle	28 mW	0%
Wearable device	**0.72**Wh	100 ms sample rate	16.5 mW	63%
		500 ms sample rate	12.48 mW	41%
		Idle	1.32 mW	0%
Smart pad	∞	Fast sample rate	2.1 W	70%
4x6	18Wh	Slow sample rate	1.85 W	50%
		Idle	1.1 W	0%
Camera	∞	High image resolution	3 W	84%
[64][46]	23Wh	Low image resolution	2 W	$\simeq$ 50%
		Idle	1 W	0%
Microsoft Kinect	∞	High resolution	12 W	91%
[92]		Low resolution	7 W	83%
		Idle	1 W	0%

Table 4.1: SAFER IoT Devices' Configurations

experiments is the Ti Sensor Tag, a BLE (Bluetooth Low Energy) dongle, that transmits its sensor data to a local controller (mobile phone) at fixed intervals, the fastest interval being 100ms. The tag uses KXT J9 accelerometer and a standard CR2032 coin battery, which has the typical capacity of 240mAh. Therefore, we do not expect the Ti Sensor Tag to work longer than 48 hours in its maximum mode. The smart pad consists of 24 FSR pressure sensors with the fast sampling rate of 5 system snapshots at every 2.4 sec.

We also measure the energy consumption of occupancy sensors as we use them in the activity recognition phase. For many years, occupancy sensors, such as motion, door and temperature sensors, have been used in smart buildings, often for lighting and HVAC control. Studies have shown that adding occupancy sensors to control lighting consumption can reduce lighting energy use from 10% to 90% or more depending on the use of the space. One study conducted on a university campus found that installing occupancy sensors to control lighting in more than 200 rooms in 10 buildings provided an annual cost savings of about $14,000 with a simple payback of 4.2 years.

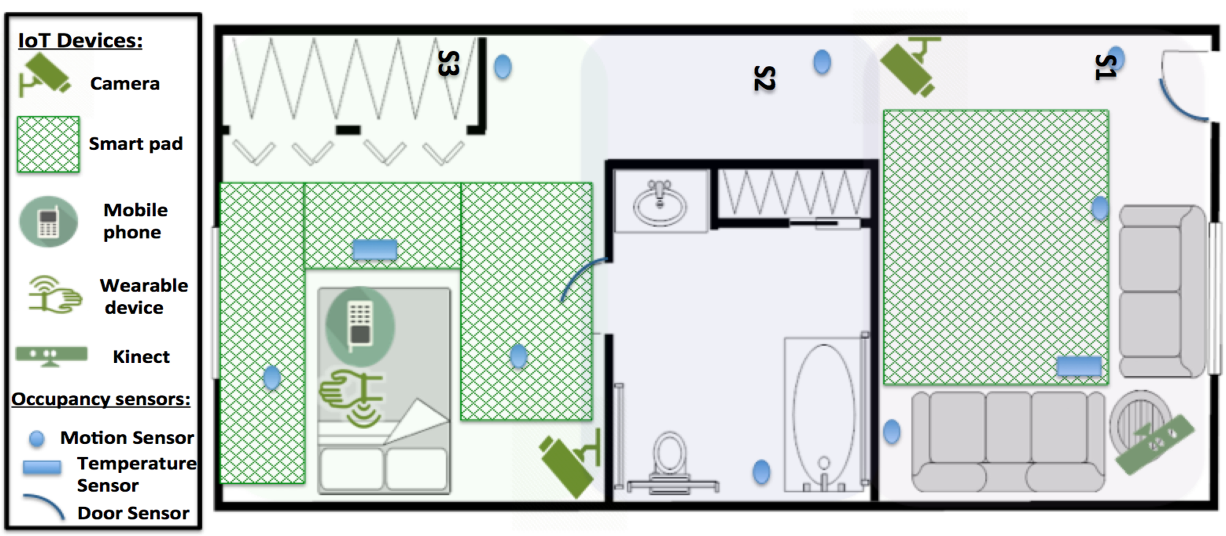


Figure 4.3: Floor-plan of the “medium” scale setting 350 ft^2, 7 IoT devices

In our large setup, as shown in Figure 4.4, we consider 39 occupancy sensors that have been used by CASAS dataset [30, 28] for daily activity recognition. Our dataset and recognition

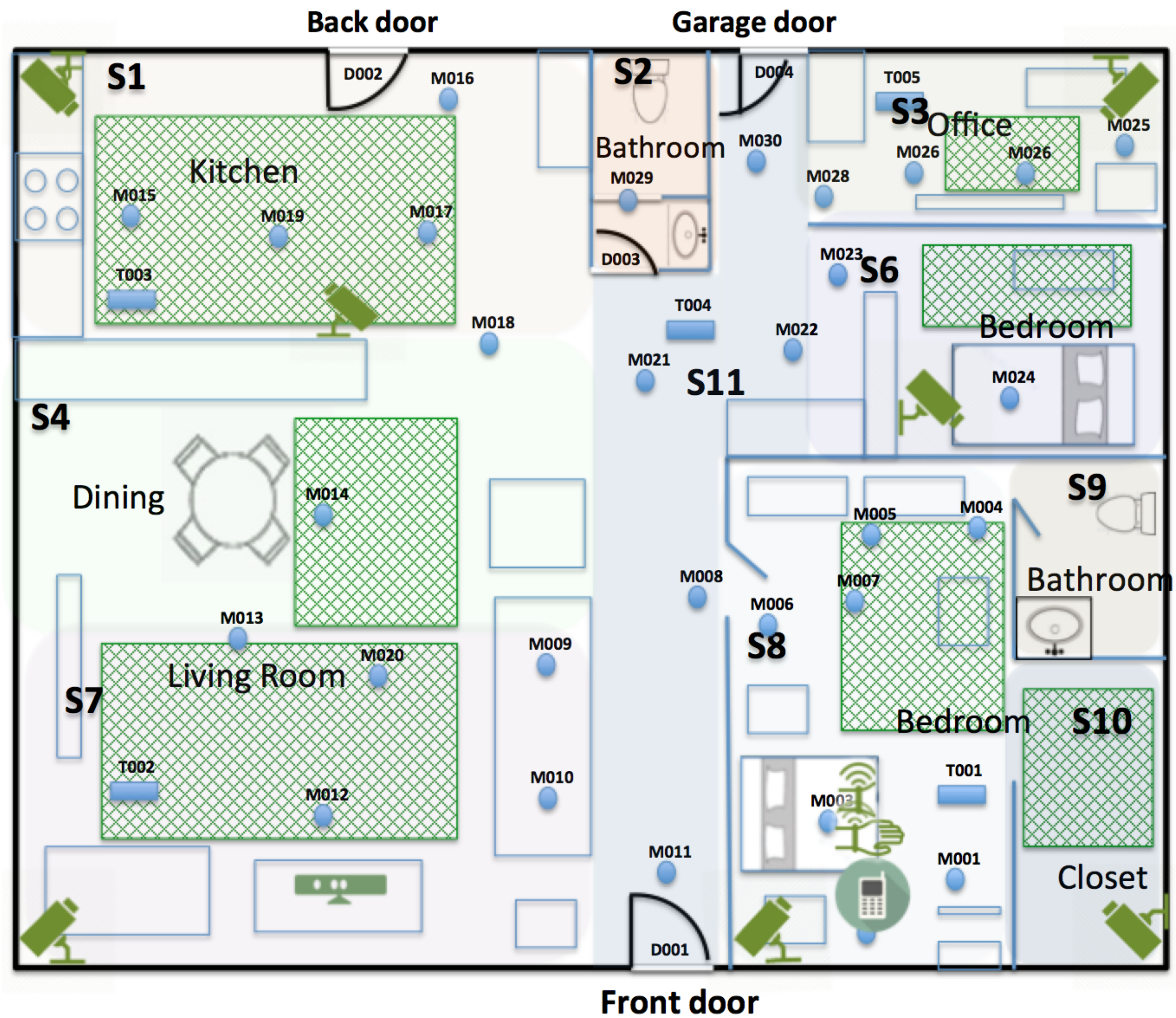


Figure 4.4: Floor-plan of the "large" scale setting 900 ft^2, 18 IoT devices

techniques are applied for one resident [28]. It comprises of 31 motion, 4 door, and 4 temperature sensors. These sensors consume power at a rate less than 8 watts and can be wired or wireless with a slow-battery-drain. In contrast, in-home fall detection systems use IoT devices, such as cameras, Microsoft Kinect, and mobile phones that consume considerably higher amount of energy. The system also contains mobile and wearable sensors that are power restricted and are fast-battery-drain devices.

4.6 Performance Evaluation and results

We have implemented five different IoT devices activation scheduling algorithms and compared their performances: all Devices, Priority based on location scheduling, Priority based on power supply scheduling, BRBL (Balanced Remaining Battery Lifetime), and CFG (Cost Function Gradient).

4.6.1 Experimental Setup - Datasets Simulation Studies

To model individuals activities in an independent living facility we leverage Aruba dataset from WSU CASAS smart home project [28, 29]; it contains sensor data and annotated activities that was collected in the home of a volunteer adult.

To conduct further experiments, we developed a discrete-event simulator and created based on real-world elderly living options four test cases at different scales in terms of size and battery devices ratio. The first case is of medium scale and corresponds to an assisted living community. The average floor-plan includes a bedroom, living room and a bathroom with an average space of 350 ft^2. (Figure 4.3) shows the floor-plan with the associated IoT devices.

We enhanced all real-floor plans with synthetic IoT devices. We assume an instrumentation density of 1 IoT device for every 50 ft^2 - the medium scale, therefore, has 7 IoT devices: 5 static wall-powered devices and 2 mobile battery-powered devices.

The second test case is an independent living community, which is a large scale deployment. Here, the average CASAS [28] floor-plan, includes 2 bedrooms, a living room, kitchen, office and 2 bathrooms with an average space of 900 ft^2 (Figure 4.4). In the large scale setup, we considered 18 IoT devices: 15 static wall-powered devices and 3 mobile battery-powered devices in the standard setup. In addition, we considered a semi-portable setting in which battery powered devices form about 50% of the IoT platform, (8 wall-powered devices and 10 battery-powered). Also, we considered a portable setup that relies on approximately 95% battery-powered devices, (17 battery powered and 1 wall-powered).

We execute our simulations on the CASAS trace dataset obtained from [28] that contains the activities of daily living of an individual in an assisted living setting for a week.

4.6.2 Performance Evaluation Metrics:

The following are metrics used in our evaluation.

- *Cumulative Energy Consumption*

  The total energy consumption can be used as a benchmark to evaluate the energy optimization algorithms.

- *Number of alive battery-powered devices*

  Intuitively, an optimized algorithm should maximize the lifetime of battery-powered devices by increasing their idle time and utilizing the wall-powered devices. As well as adjusting the effect of β which increases the priority of wall-powered over battery-operated devices.

- *Half-system lifetime*

  In addition, the algorithm increases the system lifetime by adding a time extension in proportion to the original system lifetime. Therefore, we defined the half system lifetime as the period time where at least fifty-percent of the battery devices are alive.

- *Accuracy*

  Moreover, increasing the accuracy is preferable which is the probability of detecting the event when the device operates in a configuration.

- *Power Consumption*

  Therefore, raising the accuracy, increase the power consumption cost in any device. An optimized algorithm should balance between these two parameters based on the application requirements.

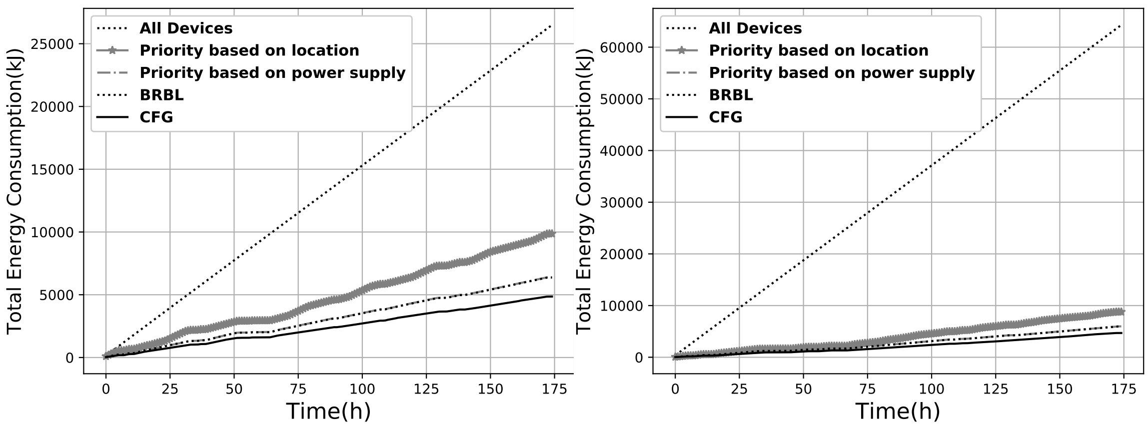
4.6.3 Experimental Results

In this section, the proposed approaches for optimizing sensing energy are validated through a detailed simulation study. We begin by comparing algorithms in terms of the cumulative energy consumption; and the battery lifetime of IoT devices. Then, we examine the effect of device density in energy consumption using the CFG algorithm and increasing β value that is related to the priority of wall-powered and battery-operated devices. Also, we show the effectiveness of different ratios of battery-operated to wall-powered devices.

1) *Basic Comparisons and Effect of Energy optimization Algorithms*

1. Cumulative energy consumption comparison:

   Figures 4.5a and 4.5b shows the total energy consumption in different algorithms/settings. As we can observe, activating sensors based on locations reduces energy consumption by half compared to the case when all IoT devices are running. The CFG algorithm saves nearly 80% to 90% energy. On the other hand, the BRBL algorithm consumes little more energy than CFG because of its high reliance on the wall-powered devices, which consume more energy than their battery counterpart. In addition, Figure 4.11 shows the gained benefit from applying CFG (Cost Function Gradient) algorithm in two settings' scale, which illustrated that the effectiveness of CFG (Cost Function Gradient) increases as the scale grows.



(a) Energy consumption in 'medium setting' (b) Energy consumption in 'large setting'

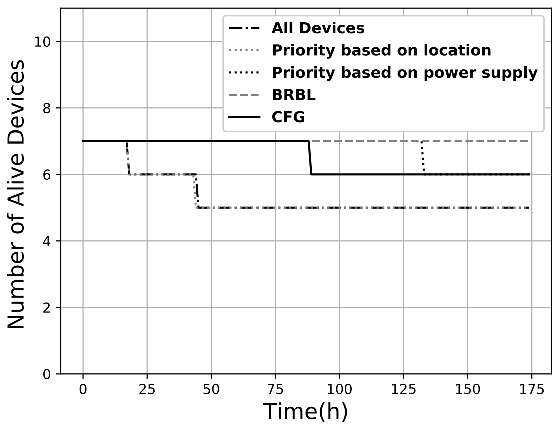
Figure 4.5: Comparison: Cumulative energy consumption

2. Battery lifetime of IoT devices

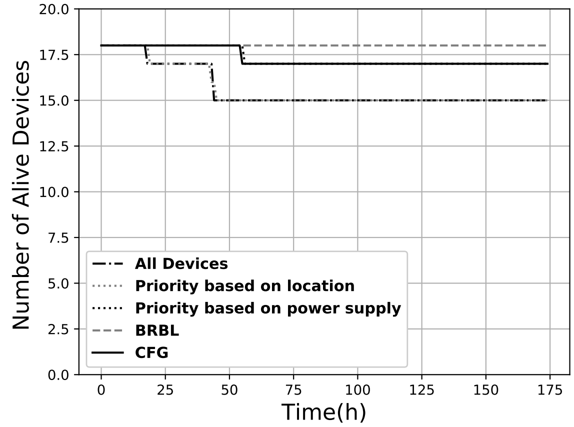
   In our experiments, we used at least two battery-powered devices: the mobile phone, and wearable devices. In our standard medium experiments, Figure 4.6a shows power drain patterns of the battery-powered IoT devices. As such, we do not expect any wearable IoT device, such as the Ti Sensor Tag, and the mobile phone to work longer then 48 hours, 20 hours respectively. Note that battery-powered devices may be occasionally recharged, but for consistent comparison we show the results for only one full cycle (from a fully charged battery to zero-power).

   The CFG algorithm extends the battery capacity 2–4 times by reducing the energy consumption of the IoT devices by choosing the best configurations for the devices. As we can see, the first IoT device in the CFG algorithm drains out after 80 hours, while the second IoT battery capacity extends to more than 175 hours.

   The BRBL algorithm extends the battery capacity of the constrained IoT devices up to 175 hours because it keeps them on the lowest configurations at all time, except in locations where wall-powered devices are not available, such as in the bathroom in the medium scale deployment.



(a) Number of alive devices in 'medium setting'

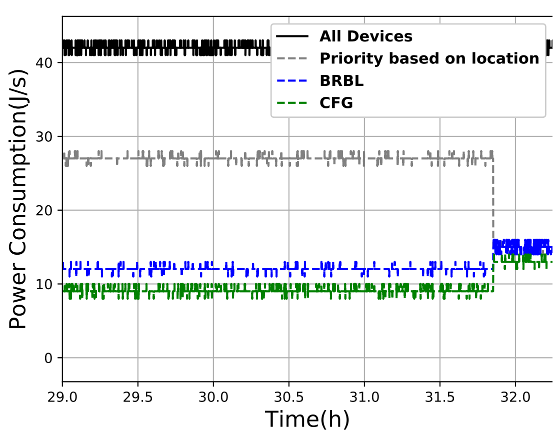


(b) Number of alive devices in 'large setting'

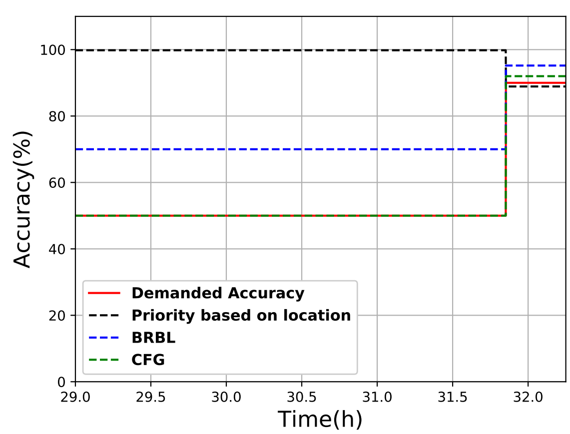
Figure 4.6: Battery lifetime of IoT devices in two settings

3. Power consumption and accuracy comparison

   Figure 4.7a and Figure 4.7b show the comparison among different algorithms in terms of power consumption and accuracy respectively. Specifically, we zoomed a 4-hour window after 29 hours to observe the difference among the different algorithms. We notice that CFG has the lowest power consumption and it remains closest to the demanded accuracy. This is because of its utilization of different configurations in different devices.



(a) Power consumption comparison



(b) Accuracy comparison

Figure 4.7: Power consumption and accuracy comparison results

2) *Scalability Comparisons and Effect of Tuning CFG Algorithm*

1. Scalability

   We examine the effect of device density in energy consumption using the CFG algorithm. In our setting, we deploy one IoT device per 50 ft^2. We double and triple this number to see the effect on energy consumption. Results in Figure 4.8 show that CFG works well when we increase the device density. As the density grows, the energy consumption also slightly rises, which is due to the energy consumption of the added IoT devices in their idle configurations.

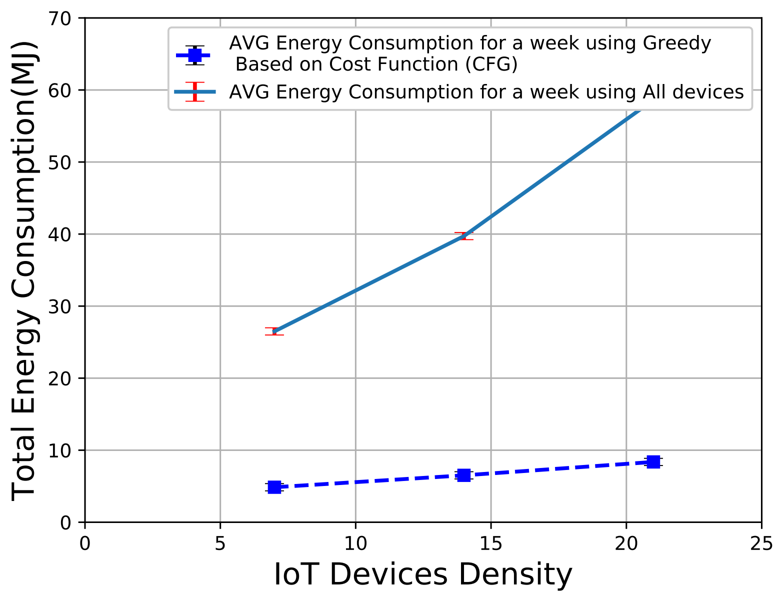


Figure 4.8: Density of IoT devices and energy consumption using the CFG algorithm in 'medium setting'

2. β and Battery lifetime of IoT devices

   We examine the effect of adjusting β in our large settings semi-portable, 50% battery-operated devices, using the CFG algorithm. Results in Figure 4.9 show that as the value of β grows, the battery-powered devices' lifetime extended. This is because when we increase β value, it increases the priority of wall-powered over battery-operated devices.

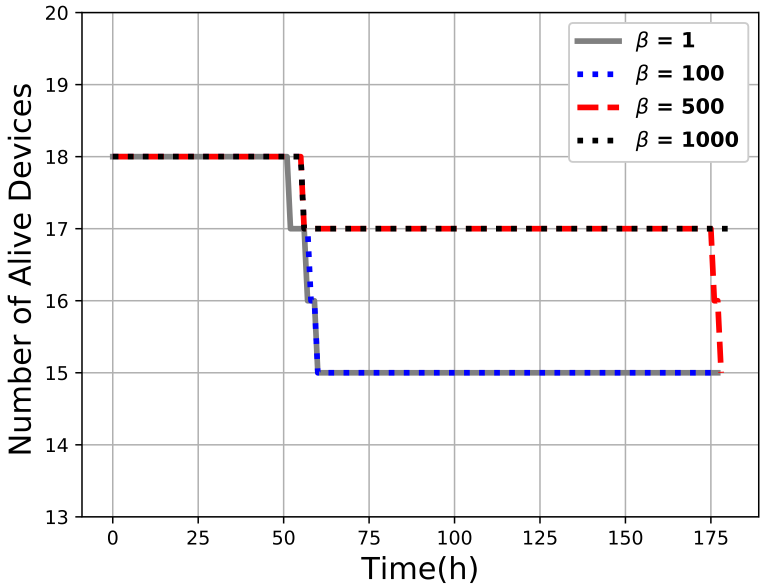


Figure 4.9: β and battery lifetime of IoT devices using the CFG algorithm in 'large semi-portable setting'

3. Half-System Lifetime

   We examine the effect of different ratios of battery-operated to wall-powered devices in our large settings (standard, semi-portable, and portable) using the CFG algorithm. Results in Figure 4.10 show that CFG works well and increase the system lifetime in all our settings. However, as the percentage of battery devices grows, starting from standard setting to portable settings the time extension in proportion to original system lifetime slightly decrease, which is due to the increase relaying on battery-powered devices.

linewidthlinewidth

Figure 4.10: Half-System lifetime in multiple setting with different ratios of battery-operated to wall-powered devices

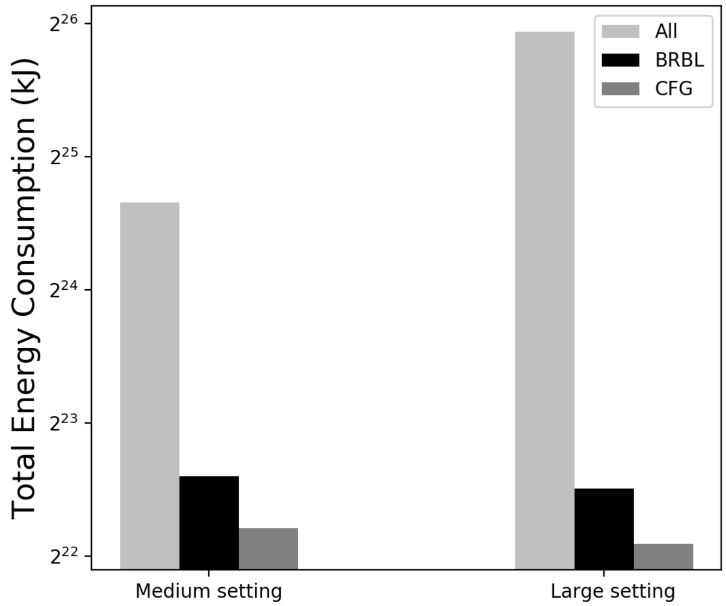


Figure 4.11: Algorithms' total energy consumption for one week in two setting using BRBL and CFG algorithms

4.7 Chapter Summary and Discussion

In this work we uniquely leveraged the concept of activities of daily living (ADLs) for energy-optimized sensor activations to create, a perpetual IoT awareness platform. We developed and deployed an elderly fall detection system; testbed measurements were used to drive larger scale simulation studies. Experimental studies with real world trace datasets indicated that the proposed Cost-Function-Gradient algorithm was able to achieve more than 80% reductions in energy consumption, doubling the system-lifetime. We believe that such techniques are essential to creating deployable IoT for mission-critical societal applications that require perpetual operations such as healthcare and assisted living.

Chapter 5

Data-Aware Multi-Network Provisioning

In this chapter, we expanded our exploration of optimization techniques for perpetual IoT systems to address other systems layers in particular – the communication/networking layer. According to a recent IEA report [78], the total additional energy consumption that results from connecting devices to a communications network is expected to increase from 500 TWh in 2010 to 1,150 TWh by 2025. Note that this projected number is for communication energy alone and does not include the energy use of the device/equipment operation. These perpetual IoT systems has multiple sensing and (sometimes multiple) network interfaces opportunities, such as Wi-Fi and LTE, also, recent advances in IoT communication technologies have created a variety of low-power connectivity options, such as NB-IoT, LTE-M, LoRa, Zigbee, etc., with different network characteristics. Activating all sensing simultaneously and send data through the interface with maximum bandwidth to achieve the best possible accuracy is inefficient and resources-intense especially for energy. Thus, in this stage our goal is to provision network resources efficiently (access networks, device capabilities) to reduce energy overhead while ensuring application quality, such as accuracy and latency.

5.1 Chapter Overview

In this chapter we aim to optimize the energy consumption of IoT-based perpetual systems in smart space settings by reducing the overhead of sensing/communication energy cost (of mobile and insitu IoT devices) while supporting essential QoS of running applications. Our key idea is to exploit the heterogeneity of IoT device, divers networking technologies, and the dynamic space states to create a context-aware energy-efficient IoT system without loss of service quality.

In this chapter, we address:

1. Modeling data collection and communication need of dynamic smartspace applications using a novel space-state abstraction.

2. Formalizing energy-efficient multi-network IoT provisioning with space-states as a constrained optimization problem, shown to be NP-hard.

3. Implementing near-optimal algorithms for IoT provisioning that leverages semantics of tasks and spaces. The phased approach involves floor-plan segmentation, space-state classification based on events, and energy optimization; exploits heterogeneous IoT device/ network.

4. Validation of our approach using a real prototype testbed for assisted living called SAFER[10]

5. Conducting an in-depth measurement study to characterize accuracy/energy cost trade-offs with multiple devices and network interfaces, that are used to drive emulated scenarios in assisted living.

6. Extensive evaluations to study the scalability and effectiveness of our algorithms/approach using simulation studies using real assisted living layouts.

5.2 Approach Overview

In this section, we describe the unique aspects of energy optimization in the perpetual IoT multi-network setting and design a space-state strategy. We characterize the energy efficiency problem in the indoor context where IoT devices are scattered in a home/building to enable indoor event detection tasks. Specific sensors include wearable devices with motion/accelerometers/GPS, ambient sensors to capture environment such as humidity/temp/gas and audiovisual sensors that capture voice/video etc. These IoT devices have varying capabilities: processing, type of power source- battery or wall-powered, reliability, accuracy, as shown in table 5.1, and are interconnected using diverse (sometimes multiple) network interfaces. Different communication technologies have different data transfer throughputs, consume different amounts of energy, and transmission range.

Recent advances in IoT communication technologies have created a variety of low-power connectivity options, such as NB-IoT, LTE-M, LoRa, Zigbee, etc, with different network characteristics. NB-IoT and LTE-M are recent protocols for low bandwidth cellular communications that connect inexpensive internet devices with small data transmission rates and smaller power consumption, i.e. higher battery life [68]. Consequently, IoT devices are today equipped with multiple network interfaces that can be customized and configured to support tradeoffs - e.g. larger datarates for communication energy consumption based on usage needs, e.g. Mobile phones(LTE, Wi-Fi ,Bluetooth), Multi-Standard SensorTag Kit (Bluetooth low energy, 6LoWPAN and ZigBee).

There are many reasons to have multiple network interfaces in those IoT devices as the produced data can be used in diverse IoT use cases, and applications; each presents unique requirements for bandwidth, range, and other connectivity features. Another reason is to increase the viability of the IoT device in the market, so if one of the interfaces becomes obsolete we still can use this device. Also, increase the efficiency in terms of energy as communication power consumption correlates to data usage. In addition, It can increase the availability of the data as there will be more than one option to send the data.

This feature is handy with perpetual services to detect critical anomalous events such as elderly falls; event likelihoods can help us tune the choice of sensors, data rates and network interfaces for better situational awareness. Table 5.1 depicts variations in service accuracies and data rates required that fluctuate based on the current activity performed in the monitored space. These measurements can guide the choices made to achieve the highest accuracy possible while ensuring energy efficiency.

IoT Node Sensors /Network Interfaces/	**Battery capacity**	**Configurations Examples**	**Power consumption**	**Performance accuracy**
Mobile phone		All sensors/Wi-Fi	2249 mW	70%
Microphone	**8**Wh	Microphone/Wi-Fi	1920 mW	45%
Accelerometer		Accelerometer,Gyroscope/Bluetooth	1172 mW	60%
Gyroscope		Accelerometer/Bluetooth	1050 mW	40%
/Bluetooth, Wi-Fi, LTE/		Idle	28 mW	0%
Wearable device	**0.72**Wh (CR2032)	All sensors/BLE	226.5 mW	40%
Accelerometer	**3**Wh (2XAAA)	Accelerometer/BLE	112.48 mW	30%
Gyroscope /BLE, Zigbee/		Idle	1.32 mW	0%
Smart pad		All sensors/Wi-Fi	3.1 W	90%
Pressure matrix	**18**Wh	Motion, LQ camera/Bluetooth	1.74 W	75%
Motion	∞	Motion, Pressure matrix/Bluetooth	1.32 W	70%
Acoustic sensor		Acoustic/LTE-M	740 mW	45%
Camera /Bluetooth, Wi-Fi,		Motion/NB-IoT	650 mW	20%
NB-IoT, LTE-M/		Idle	330 mW	0%

Table 5.1: IoT Configurations in the SAFER Prototype

For example, in the SAFER assisted living context[2], activating all platform IoT nodes for a fall detection service will offer above 90% accuracy. This high level of accuracy is wasteful at times of low activity (during night time) when lower levels of sensing/transmission are adequate. However, higher accuracy is required during the wake-up times; studies indicate that 70% of elderly falls happen during this time. This illustrates the need to provide abstractions that capture the dynamicity of the underlying space into multiple modes that can then be used to trigger increased sensing based on event shifts.

We introduce a key abstraction for exposing indoor space dynamicity in IoT deployments to build more reliable and energy-efficient systems. For this, we observe that there are three key modes that the physical space shifts between based on the occurrence of events, We refer to these settings as *space-states*. We classify physicalspaces as being in one of three space-states **Normal, Anomaly, and Emergency** modes that capture different conditions and vary in the amount/quality of data that needs to be sensed and transmitted. Ambient sensing at low datarates occurs in the *normal* mode. For example, this allows upon sensing a potential gas leak, the system switches from "normal" to the "anomaly" space-state; more data to identify the causes and actions. A latency tradeoff occurs to capture more information in the anomaly mode. In a fire event (smoke, heat, gas detection), the space-state shifts to the emergency state where low latency and high data quality are needed - we enable more sensors and high bandwidth interfaces to detect the causes of fire/recommend evacuation routes[74].

In each mode, knowledge of device capabilities can be used to activate an adequate subset of data sources to meet the accuracy/latency levels based on the space-state. Different connectivity options have varying characteristics and energy cost that can be leveraged as Wi-Fi module is the most energy-hungry [38], this can be utilized in such a dynamic space to activate the network interface that accommodates the streaming data rate. Battery-powered devices such as mobile/wearables dissipate power quickly and need to be recharged. Furthermore, we observe that one can designate areas in the floor plan to be used to identify event patterns corresponding to ADLs; i.e. not all wall powered IoT devices need to be activated all the time. Knowledge of the space-state and activities of daily living (ADLs) of a resident can provide us with semantic information about the location, activity type, duration, etc.; this can be utilized intelligently to minimize energy dissipation in the integrated system.

Given the above observations, our goal is to minimize energy consumption of the combined (network and IoT device) deployment to enable long-term operation while meeting accuracy threshold demands.

5.3 The Energy Optimization Problem for Multi-Network IoT Platforms in Smart Spaces

In this section, we first introduce various terms comprising of our problem (the IoT multi-network energy problem) then discuss our assumptions about the system; this is followed by a detailed formulation of device-network activation as an optimization problem.

5.3.1 Terms and assumptions

We assume that in each home/building there is a *local controller* that is capable of connectivity with multiple devices and their networking- this controller maintains state info. and keeps track of nodes and networks. Each node has one to many network (*interfaces*) to communicate and send their data. Each node comprises one to many sensors (*data sources*) that feed data to a designated service. We also assume that each node can choose at most one interface at any given time out of its available network interfaces.

We assume *applications* are part of smart building/home systems that deliver multiple *services*. For example, a smart meeting application (e.g., the application Noodle in the smart building platform TIPPERS [3]) can deliver services such as recording audio/video and speech recognition. In general, data out-of a set of sensors from a given set of nodes are required for a service to be realized. Consider the fall detection service for an elderly individual- data from floor mat sensors and cameras around the subject's current location are required;

other sensors need not to be activated.

We also classify nodes into two groups in terms of their source of energy/power as: unconstrained *wall-powered devices* and constrained *battery-powered* devices. Obviously, a trade-off exists between the energy consumed by a node and its different network configurations, bandwidth vary and latencies as well. Typically, higher bandwidth interfaces (e.g., Wifi) are desirable if the data volume is high; however, such interfaces are often energy-consuming. When possible, lower bandwidth interfaces are preferred since this increases system lifetime. Hence low latencies imply high energy consumption and shorten system lifetime, sending an adequate amount of data through an appropriate network interface results in an energy-optimized sustainable IoT system.

5.3.2 Problem Statement and Formulation

We formulate energy optimization for IoT multi-networks as a constrained optimization problem. Let us denote I as the set of all candidate nodes (indexed by i) in a certain segment/building at a given time, J be the set of all types of data sources/sensors in the system (indexed by j), and K be the set of all available interfaces (indexed by k). Obviously, not all nodes will have all sensors, nor all interfaces. We denote a binary indicator p_{ij} to denote if node i contains data source type j onboard, s_{ij} to denote if data source j at node i is required for the service at hand, and q_{ik} to denote if node i has interface k onboard. Given these, we are required to compute two sets of binary decisions: (i) x_{ij} indicating if data source j on node i must be selected for the service, and (ii) y_{ik} indicating if node i chooses interface k to transmit data at that time interval. The objective is to minimize the overall energy consumption subject to latency and accuracy constraints. This selection of sensors/interfaces happens periodically at an interval denoted by T.

Each sensor type j has the following attributes: the data generation rate (denoted by dr_j, measured in bytes per sec) and the energy consumption rate, e_{ij}, to indicate the rate at which energy gets depleted when the sensor gets activated at node i. Admittedly, different data sources such as video, audio, and motion generate data in different rates and consequently consume different amount of energy. Again, a data source on a certain node has predetermined accuracy for a service, denoted by a_{ij}. Finally, each interface k has the following attributes: energy consumption rate (e_k, measured in Joules per byte), bandwidth of the interface (bw_k, measured bytes per second), and the propagation delay/latency of the interface (pl_k). Note that the propagation delay accounts for the signal propagation delay from the transmitter to the receiver, but the actual end-to-end delay (latency) depends on, in addition to the propagation delay, to the volume of data being sent over the interface, which in turn depends on which data sources are chosen and the bandwidth of the interface. Consequently, the effective end-to-end latency l_{ik} (at node i for sending data on interface k) is given by:

$$l_{ik} = pl_k + \left(T \cdot \sum_j x_{ij} dr_j\right) \cdot \frac{1}{bw_k} \tag{5.1}$$

As noted earlier, the service in question has two constraints: a *latency constraint* and an *accuracy constraint*. The latency constraint dictates that the end-to-end latency of collecting data from all of the chosen data sources and should not exceed a certain bound. This bound is called the *latency demand* and is denoted by τ. Interestingly, the latency demand can be a variable (rather than being a constant) that may change over time depending on the service operation and its space state. For example, τ can take a lower value for a time-critical service (when quicker responses are demanded) than a normal service when the latency demand can be relaxed (by setting τ to a higher value).

The second constraint, the accuracy constraint asks for certain accuracy of the service. We argue that when data from multiple sources are combined for a given service, the accuracy of the service should increase. By defining accuracy as the probability of detecting some event of interest, we can use a soft OR operator to combine accuracy when data from multiple data sources are utilized as follows: if one source yields accuracy a_1 and another source one yields a_2, the combined accuracy is given by: $1 - (1 - a_1)(1 - a_2)$ (probability that at least one of the two sources detects the event), or more generally, $1 - \prod_m (1 - a_m)$ for relevant m's.

The accuracy constraint dictates that the combined accuracy over all collected data from the chosen data sources should exceed a threshold, called the *accuracy demand* (denoted as α). Like the latency demand, the accuracy demand can also be a variable that may change over time depending on the service space state. For example, when the service detects an anomaly, the accuracy requirement becomes higher compared to when the service was running as normal.

As stated above, our objective is to minimize the overall energy consumption for collecting and sending data over the interfaces across all nodes. We also want to extend the lifetime of battery-powered devices, so we consider the remaining battery capacity of those devices. We define *energy-cost*, denoted by c_{ik}, for each interface choice per node as follows:

$$c_{ik} = T \cdot \eta_i \sum_j x_{ij} \left(e_k \times dr_j + e_{ij} \right) \tag{5.2}$$

The cost is proportional to the amount of energy consumed by the interface, which is e_k times the data volume generated by the sensors chosen at the node, combined with the amount of energy consumed by the node for activating the sensors (e_{ij}'s). The cost also takes into account the fact that operating a battery-operated device is costlier than an equivalent wall-powered device, when they both consume the same amount of energy. The operation arguably gets costlier when the remaining battery capacity becomes low. To reflect this, we multiply the base energy consumption with an adjustment factor, η_i, which is given by (the expression is adopted from [10, 11]):

$$\eta_i = 1 + \beta \cdot \exp\left(-\frac{r_i}{r_i^0}\right) \tag{5.3}$$

where r_i^0 denotes the initial battery capacity of the node, r_i is the current remaining energy, and β is a tune-able parameter to adjust the effect.

We, therefore, have the following optimization: find x_{ij} and y_{ik} so as to—

$$\text{minimize} \quad \sum_{i \in I} \sum_{k \in K} y_{ik} \cdot c_{ik} \tag{5.4}$$

$$\text{subject to} \quad \sum_{i \in I} \sum_{j \in J} x_{ij} \cdot \log(1 - a_{ij}) \leq \log(1 - \alpha) \tag{5.5}$$

$$y_{ik} \cdot l_{ik} \leq \tau, \forall i, k \tag{5.6}$$

$$y_{ik} \leq \sum_{j \in J} x_{ij}, \forall i, k \tag{5.7}$$

$$\sum_{k \in K} y_{ik} \leq 1, \forall i \tag{5.8}$$

$$x_{ij} \leq p_{ij} \cdot s_{ij}, \forall i, j \tag{5.9}$$

$$y_{ik} \leq q_{ik}, \forall i, k \tag{5.10}$$

$$\forall x_{ij}, y_{ik} \in \{0, 1\}, \forall i, j, k$$

The objective (5.4) captures how to minimize the total energy cost for all nodes for the chosen associated interfaces. Eq 5.5 is actually a rewritten expression (taking the log of both sides with some adjustments) of the following equation:

$$1 - \prod_{i \in I} \prod_{j \in J} (1 - x_{ij} a_{ij}) \geq \alpha \tag{5.11}$$

This ensures that the effective accuracy accumulated over all data sources remains higher than the accuracy demand, α. Eq 5.6 aims to meet the latency constraint so that the latency of *all* data sources remains within τ. Eq 5.7 ensures that an interface on a node is activate (at most once, Eq 5.8) only if some sensors on the node is chosen. The last two constraints allow only onboard data sources and interfaces to be chosen for a given node.

The IoT MULTI-NETWORK ENERGY OPTIMIZATION PROBLEM as formulated above is an NP-hard problem that can be reduced from the minimum multidimensional multiple-choice knapsack problem [53] as shown below.

Looking closely at the formulation, it reveals that for each node, we are required to find two things: the subset of data sources to be chosen (x_{ij}) and the network interface to be used (y_{ik}). We can combine these two selections into one as follows. For each node, we construct a list of *choices* where each choice is a tuple in the form of (S, k) where S is some *subset* of data sources for some choice of interface k. Ideally, this list may contain all possible combination of choices over the set of data sources and interfaces that the node has. But the choices for which the associated latency exceeds τ can be treated as invalid because they violate the latency constraint and hence can be dropped. In addition, there can be an *empty* choice denoted as $(\emptyset, \text{null})$ (no sensors and no interface are chosen).

With this transformation, we are now required to choose *exactly* one choice per node that minimizes the total energy cost subject to the accuracy constraint. This is effectively an instance of the classical Multiple-Choice Knapsack Problem (MCKP) and we use the classical MCKP heuristic to solve this. Since the number of interfaces and data sources per node, in practice, is small (in the order of 10 or fewer), the total number of choices are rather bounded and an efficient algorithm can be devised (refer to the next section).

5.3.3 Proof on computational complexity

We prove the computational complexity (NP- hardness) of the IoT multi-network energy optimization problem, by showing that the minimum multidimensional multiple-choice knapsack problem, which is known to be NP-complete, can be reduced to it. Knapsack problem has been widely studied in computer science for years. It is one of the problems on Karp's original list of 21 NP-complete problems [51]. There exist several variants of the problem, in this proof we considered the minimum multidimensional multiple-choice version of the problem.

The formulation of the energy optimization for IoT multi-network problem is:

According to our settings,we have a set of n heterogeneous IoT nodes in a certain segment, $i = 1, \cdots, n$. Each IoT node i can be described by a *profile*, which consists of multiple network interfaces k.

e_{ik} and l_{ik} denote the rate of energy consumption and latency level respectively for network interface k of IoT node i $(1 \leq k \leq K)$. In addition, each IoT node consists of multiple sensors j. a_{ij} denote the accuracy level for sensor j of device i $(1 \leq j \leq J)$. Also, each IoT node has a remaining battery capacity, denoted by r_i, at a certain time. Note that for wall-powered devices r_i is not defined or assumed to be ∞.

We prove this constraint optimization problem, formulated as equation(**??**) with all information available, is an NP-hard by showing that the minimum multidimensional multiple-choice knapsack problem which is known to be NP-complete, can be reduced to it.

In order to define the minimum multidimensional multiple-choice knapsack problem formally, consider m mutually disjoint classes $N_1 \cdots N_m$, $i = 1, \cdots, m$ of items to be packed into a knapsack satisfying the capacity constraints in each of the d dimensions. Each item j in i class has cost c_{ij}, utility u_{ij} and a size s_{ij}, and the problem is to find a subset of exactly one item from each class such that least profitable set of items such that the total size of the selected items is to be at least the capacity C. And the total utility of the selected items is to be at most the capacity U.

If we introduce the binary variables x_{ij}, which take on value 1 if and only if item k is chosen in class i, the MMMCKP can be formulated as:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{m} \sum_{j \in N_i} x_{ij} \cdot c_{ij} \\ \text{subject to} \quad & \sum_{i=1}^{m} \sum_{j \in N_i} x_{ij} \cdot s_{ij} \geq C \\ & \sum_{i=1}^{m} \sum_{j \in N_i} x_{ij} \cdot u_{ij} \leq U \\ & \sum_{j \in N_i} x_{ij} = 1, \forall i \end{aligned} \tag{5.12}$$

The IoT multi-network energy optimization problem belongs to NP as there is a subset of IoT nodes with selecting at most one network interface k from each of n IoT nodes, each node i has K different network interfaces, that has the least total cost, and the total combined-accuracy is more than or equal to the service's accuracy demand α and the max latency is less than or equal to the service's-state latency demand τ. Thus, if we have a proposed correct "yes" solution, we can verify this solution in polynomial time $O(n)$ by checking that energy optimization for IoT multi-network problem has a satisfying subset.

Reduction from minimum multidimensional multiple-choice knapsack problem $<_P$ energy optimization for IoT multi-network problem. In other words, minimum multidimensional multiple-choice knapsack problem is polynomial reducible to the energy optimization for IoT multi-network problem. We consider an instance of minimum multidimensional multiple-choice knapsack problem, and we will construct an equivalent instance of the energy optimization for IoT multi-network problem.

- m mutually disjoint classes in minimum multidimensional multiple-choice knapsack problem $\rightarrow$ n IoT nodes each with multiple network interfaces in IoT multi-network energy optimization problem.

- Class N_i has multiple j items $\rightarrow$ IoT node i has K different network interfaces.
- c_{ij} cost for each item in class N_i $\rightarrow$ c_{ik} cost for each IoT node i in its network configuration k.
- s_{ij} size of each item $\rightarrow$ (-ln$(1 - a_{ij})$) accuracy delivered by each IoT node i in its configuration k, note that this is $(-\ln(1 - a_{ik}))$ a positive value as $0 \leq a_{ij} \leq 1$
- C the least capacity limit $\rightarrow$ (-ln$(1 - \alpha)$) demanded accuracy.
- u_{ij} utility of each item $\rightarrow$ l_{ij} delay by each k interface.
- U the most utility limit $\rightarrow$ τ acceptable latency.

Therefore, the minimum multidimensional multiple-choice knapsack problem can be reduced to IoT multi-network energy optimization problem, which means energy optimization for IoT multi-network problem is at least as hard as the minimum multidimensional multiple-choice knapsack problem. Therefore, the IoT multi-network energy optimization problem is NP-hard.

5.4 Algorithms and Heuristics for Energy Optimization in Multi-Network IoT Platforms

In the following sections, we will propose a description of set of feasible design intuitions towards practical algorithms to optimize energy efficiency in IoT platforms over heterogeneous networks.

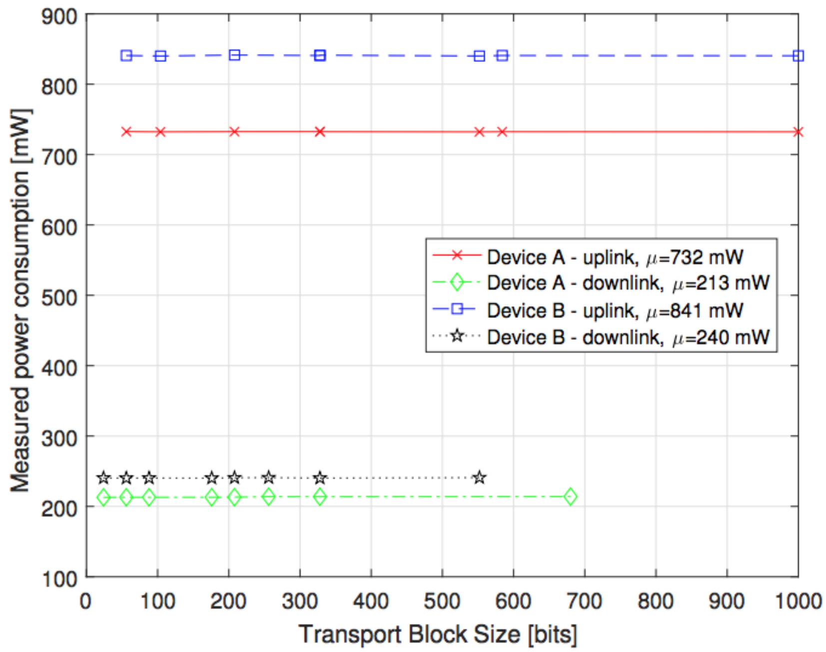


Figure 5.1: Nb-IoT the data rate does not directly impact the power consumption [5]

5.4.1 Design intuitions

We have considered the following intuitions in the design of our solutions that take into account space-related features:

Location-aware space segmentation: Here the goal is to segment regions of the home into space partition that are semantically meaningful. This gives us a viable strategy to drive system energy efficiency, scalability, and performance.

Exploiting space-state modes: We determine a baseline level of sensing in a space that is low-cost in terms of resources for energy, communication, etc. to detect ambient events/activities. Mapping events in a space and their urgency to sensors can help classify level of sensing needed (normal, anomaly, emergency).

These base-line sensors can be wired or wireless and consume power at a very low rate, such as less than 8 watts consumption in layout of the 39 sensors (31 motion, 4 door, and 4 temperature) deployed on the 900 ft^2 home, as shown in Figure 5.4; that have been used for daily activity/space-state recognition [30, 28].

5.4.2 Framework and Algorithms

Developing an optimal energy efficient system for perpetual and heterogeneous IoT operation needs comprehensive knowledge about the floor-plan architecture, space-state, semantics patterns, and IoT device profiles/status. To handle the complexity that arises due to the dynamic nature and the diversity of the underlying infrastructure, we propose a three-phase system framework that utilizes the design intuitions mentioned above as illustrated in Figure5.2.

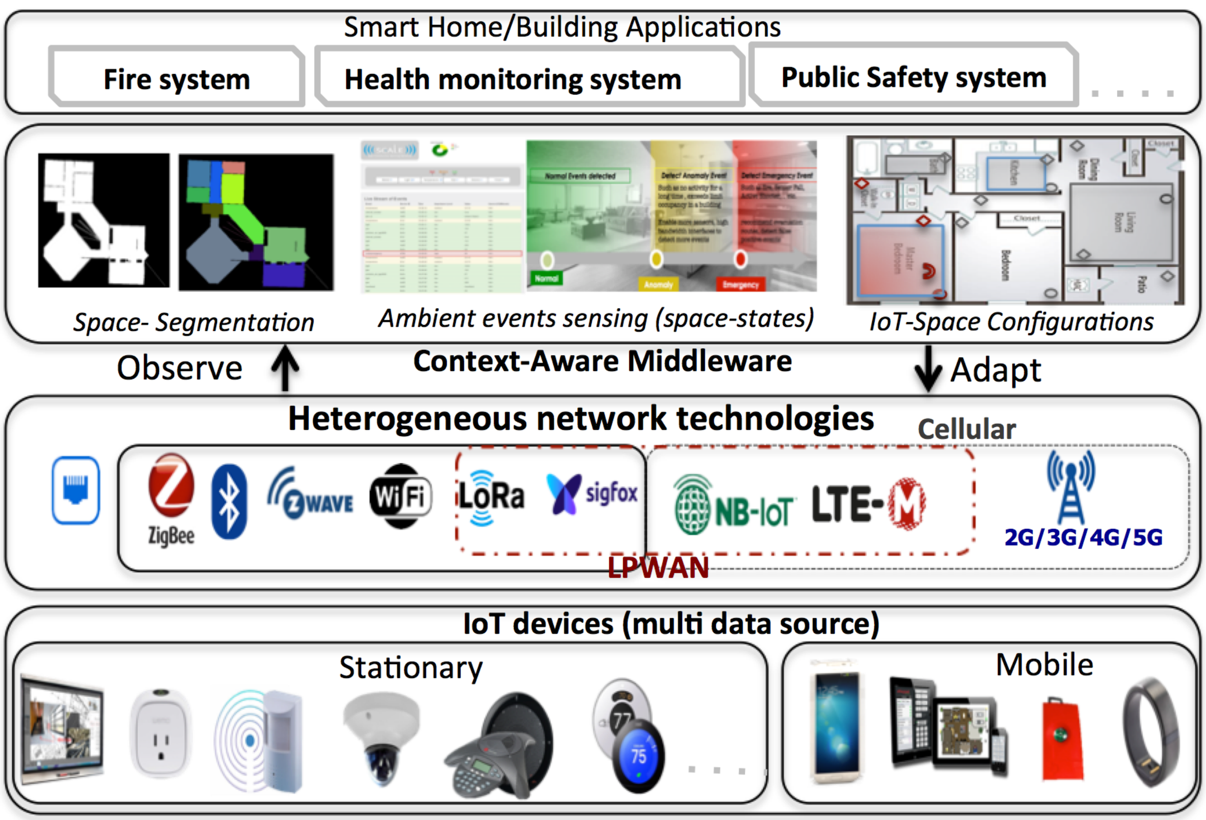


Figure 5.2: The proposed system architecture

Base-line algorithms

We propose two of base-line techniques that allow us to evaluate a family of algorithms based on space-state semantics. We activate all IoT nodes that are needed by the service at hand (e.g., fall detection), furthermore, for each node we enable all data sources (sensors) required by the desired sensing service: for instance, a fall detection service leverages the smart pad, audio/video sensing, with the highest bandwidth network interface available, Wi-Fi.

Location-Aware algorithm: In this approach, we activate the IoT nodes that are present in the segment area where the user is currently located based on analysis of data from the ambient sensing. The scheme chooses the highest data rate to get the highest accuracy with the highest bandwidth network interface available.

Context-Power-Aware algorithm: In this approach, we take into the consideration the space-states (normal, anomaly, emergency) and the demanded accuracy of operation; each space-states mode requires different level. In this technique, the platform maximizes the system lifetime by activating the wall-powered devices first, before choosing the battery-operated devices in descending order of their remaining battery capacity until we exceed the current space-state's accuracy threshold. This basic algorithm exploits space-state semantics but in a simplistic manner.

Greedy Algorithms

In this subsection, we outline greedy heuristics for the selection of data sources (sensors) and network interfaces per node. As per the formation outlined in the earlier section, we are required to find the best *choice* for each node, that is, the best (S, k) tuple per node that optimizes the total energy cost subject to the accuracy constraint. For the n-th choice at node i, we compute the following two quantities: *energy(i, n)* and *accuracy(i, n)*, the total energy cost and the combined accuracy, respectively, associated with the choice of data sources and the interface. Given these, we can construct selection techniques that yield two heuristics:

Node-approximation: For each choice (i, n), we compute the ratio of *energy(i, n)* to *accuracy(i, n)* and then rank the choices per node in ascending order this ratio (0/0 is assumed to be 0). Once the choices are ranked, the algorithm builds the solution as follows. It starts by taking the first choice (the empty choice) from each node (zero energy cost with no interface is chosen) and then progressively moves to the next *immediate* choice per node (one node at a time) and compute the combined accuracy with the associated choices across all nodes (using the soft OR operation described before). The algorithm ends when the accuracy demand is met (combined accuracy exceeds α) or the solution cannot be improved any more.

System-wide greedy: In this approach, the search iterates over the entire (i, n) space instead of doing it per node. This approach uses the classical gradient-based MCKP heuristic [90]. The choices per node are ranked in the ascending order of accuracy (for the choices having the same accuracy, only the lowest energy choice is kept). The algorithm, starting with an empty choice, makes a sequence of changes in which the current choices from each node are upgraded to the next best based on the gradient of energy cost change to accuracy change, given by the ratio $\frac{\Delta energy}{\Delta accuracy}$

The goal is to move toward the choice that offers lower change in energy cost compared to a big change in accuracy. The process continues until the accuracy demand is achieved (Algorithm 1).

More formally, the algorithm works as follows:

- Construct a list $(i, b(i))$; $b(i)$ is the current best choice for node i.

Algorithm 2: *System-Wide-Greedy*(α)

$ChoicesList_i(\{S\}, k)$ all combinations of available IoT nodes(i)with its feasible *sensors*$\{S\}$subset associated with k *interface*
$ChoiceResult_i$ Empty$_{set}$($\{\}$,null)
$CombinedAccuracy = 0$;
while ($CombinedAccuracy \leq \alpha$) **do**
    foreach i **do**
        $\Delta EnergyCost = ECost_{next}(i) - ECost_{current}(i)$
        $\Delta Accuracy = Accuracy_{next}(i) - Accuracy_{current}(i)$
        $CalculateSlop(\{i\})(\Delta EnergyCost/\Delta Accuracy)$
    end
    Select i with the largest slope;
    $Choice - Result_i = Choices - list_i(next)$;
    $CalculateCombinedAccuracy$;
end

- Initialization: set $b(i) = 0$ (the empty choices).
- Set $\Delta E(i) = energy(i, b(i) + 1) - energy(i, b(i))$
- Set $\Delta A(i) = accuracy(i, b(i) + 1) - accuracy(i, b(i))$
- Compute $gradient(i) = \frac{\Delta E(i)}{\Delta A(i)}$.
- Find $i^* = \text{argmin}_i gradient(i)$.
- Update $b(i^*) = b(i) + 1$.
- Repeat until the combined accuracy $\geq \alpha$.

5.5 Performance Evaluation and Results

The perpetual IoT platform is derived from our existing community-oriented IoT deployments in SCALE [97] that was deployed in Victory Court Senior Apartments in Montgomery County, MD. Leveraging that, we developed SAFER [10], an elderly fall detection system, that helped us to explore challenges arising in real world deployments and to collect measurements varying and realistic combinations of sensors to drive our simulations.

5.5.1 Prototype Platform and Measurement Study

We aim to investigate the energy consumption of various IoT access network technologies, such as Wi-Fi, Bluetooth, NB-IoT, and LTE-M, to present their average energy consumption based on our conducted measurements and datasheets [25, 4, 5, 6, 38, 16, 52]. For this we built on the SAFER platform and expand our testbed to add new common interface technologies; *SAFER: prototype platform and testbed:* The smart pad, shown in Figure 5.3, is comprised of multiple data sources (matrix of Square Force-Sensitive Resistor sensors, motion, camera, acoustic sensor) with multi-connectivity options: Wi-Fi (Edimax EW-7811Un), Bluetooth, NB-IoT, and LTE-M (Cellular IoT Application Shield).

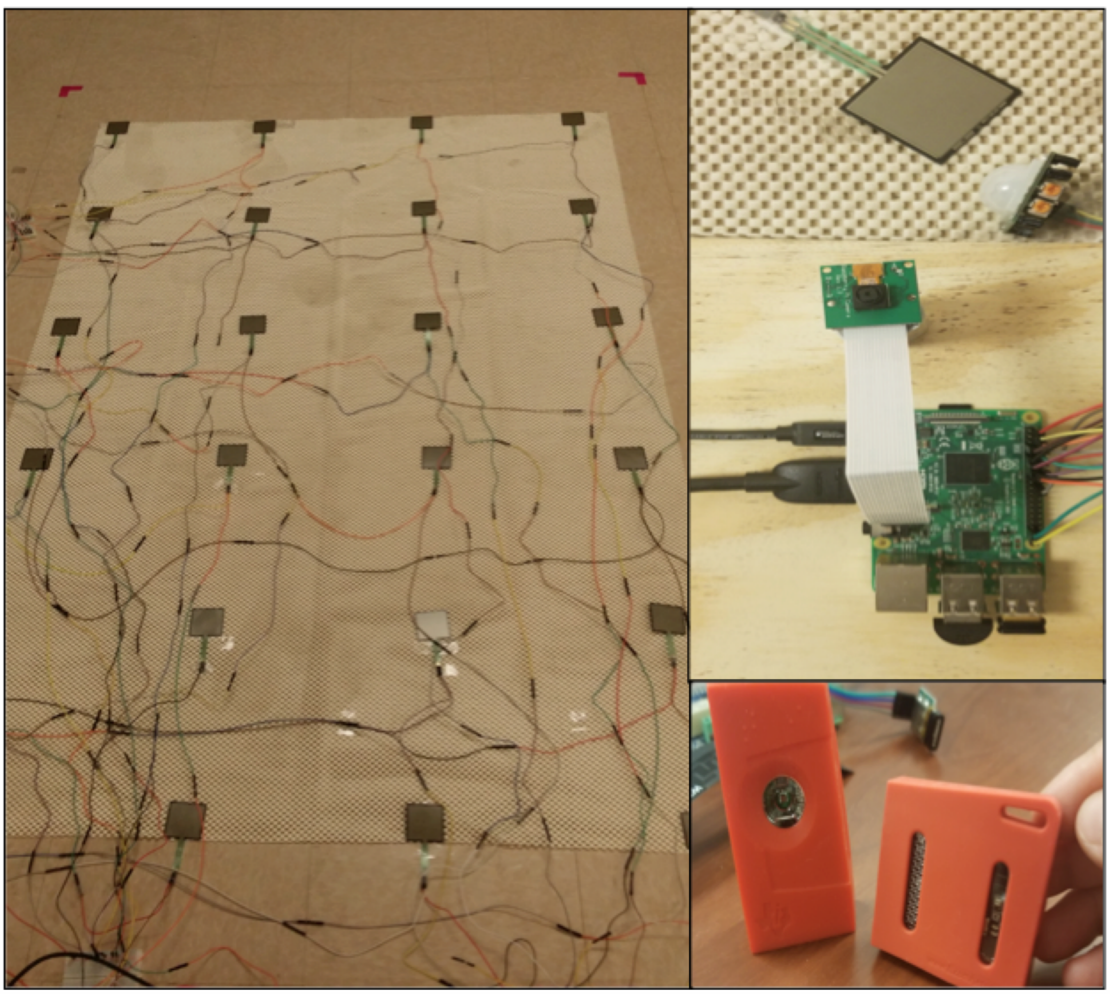


Figure 5.3: The SAFER platform devices

The wearable sensor (Ti SensorTag CC2541), Figure 5.3, incorporates up to 10 sensors: light, digital microphone, magnetic sensor, humidity, pressure, acc., gyroscope, magnetometer, object temperature, and ambient temperature. It includes interfaces BLE, 6LoWPAN and ZigBee. The setup also has a mobile phone with a set of onboard sensors. Nodes publish their data to a backend server using the MQTT protocol.

The measurements shown in Table 5.2 include the transmission range, bandwidth capacity in bits per second and power consumption. We conducted several iterations of communication measurement study using the SAFER platform we created. Figure **??** shows the visualization of our measurement setting during collection. Using a Tekpower Digital Multimeter, Raspberry Pi, Raspberry Pi Cellular IoT Application Shield – LTE-M NB-IoT eGPRS), Edimax Wi-Fi USB Adapter, Soundbot Bluetooth 4.0 USB Adapter, USB Male to DIP, USB 2.0 Female Breakout Boards, we measured communication energy cost by measuring the current while sending data through different available interfaces in our platform; in addition, we took into consideration the LTE-M NB-IoT Data Sheets.

Communication Technology	**Maximum bandwidth**	**Transmission range**	**Latency**	**Avg. Power Consumption**
Wi-Fi (+router)	54 Mbps	$<$ 300 feet	2-3ms	1800 mW
Bluetooth (+gateway)	3 Mbps	$\approx$ 300 feet	100ms	1000mW
NB-IoT (Cat-NB1)	250 kbps	$\approx$ 6 miles	1.5-10s	480 mW
LTE-M (Cat-M1)	1 Mbps	1-3 miles	50-100ms	500 mW

Table 5.2: Characteristics comparison of access network technologies that have been used in our platform [25, 4, 5, 6, 38, 16, 52]

We observe that energy cost per bit is higher if a small amount of data is sent over a higher energy interface with higher bandwidth. According to [16], Bluetooth uses nearly 3% of Wi-Fi's energy; for example, sending data at the rate of 75 bytes/sec over Wi-Fi requires 80 mW whereas Bluetooth consumes only 2mW. In NB-IoT, however, the data rate does not directly impact the power consumption [5], similar to WiFi radio [16].

We, therefore, calculate energy efficiency (in Joules per bit) of different network interfaces at various data rates, shown in Table 5.3. It is also observed that data sources have varying data generation rates (e.g., sensors like humidity/temperature/GPS produce 120-200 bps whereas video can reach up to 10Mbps).

Communication Technology	**The communication energy efficiency based on data rate**			
	100 bps	**100 kbps**	**1Mbps**	**10Mbps**
Wi-Fi (+router)	18.00 mJ/b	18.00 J/b	1.800 J/b	0.1800 J/b
Bluetooth (+gateway)	10 mJ/b	10 J/b	1 J/b	N/A
NB-IoT (Cat-NB1)	4.8 mJ/b	4.8 J/b	N/A	N/A
LTE-M (Cat-M1)	5.7 mJ/b	5.7 J/b	0.57 J/b	N/A

Table 5.3: Energy efficiency based on data rate comparison for access network technologies that have been used in our platform [25, 4, 5, 6, 38, 16, 52]

5.5.2 Experimental Setup - Simulation Studies

To conduct further experiments, we developed a fixed-time interval simulator and created multiple test cases at different scales/devices intensities based on real-world layout of elderly living options. The floor-plan includes 2 bedrooms, a living room, kitchen, office and 2 bathrooms with a space of $900ft^2$ (Figure 5.4).

With a density of 1 IoT device per 50 ft^2, we considered a total of 18 devices (15 static wall-powered devices and 3 mobile battery-powered devices).

We also considered two more settings with density 1 node per $30ft^2$ and 1 node per $20ft^2$, respectively. We execute our simulations on the CASAS trace dataset obtained from [28] that contains the activities of daily living of an individual in an assisted living setting for a week (ADLs are used to switch between space-states).

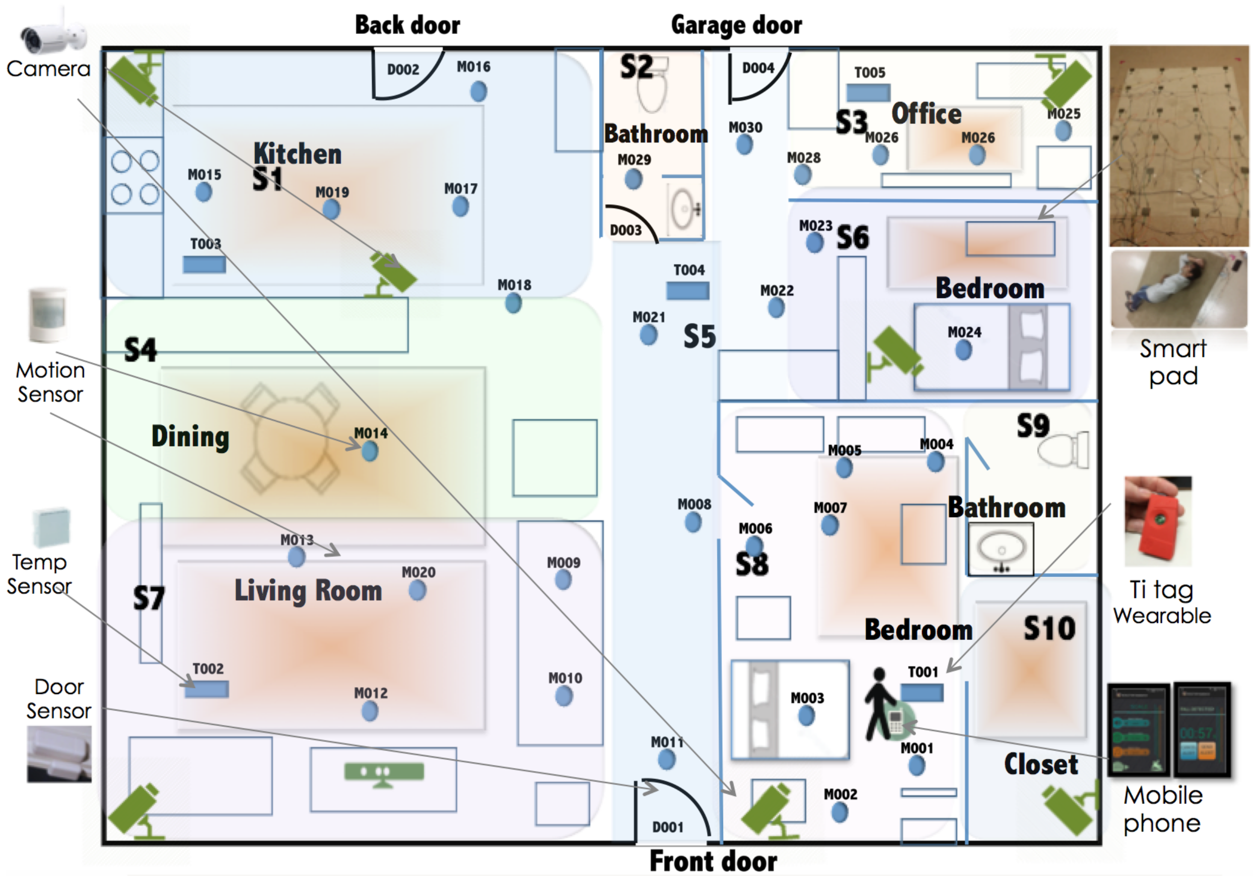


Figure 5.4: Segmented floor-plan for an assisted living setting (SAFER with base-line ambient sensing) [28]

We use the following performance metrics.

Cumulative energy consumption: the total energy consumption can be used as a benchmark to evaluate the energy optimization algorithms.

Sensing and transmission energy consumption: we examine the effect of different algorithms in the ratios of sensing and transmission energy consumption.

Number of active/idle IoT nodes: intuitively, an optimized algorithm should reduce the number of active IoT nodes while attaining the demanded accuracy.

5.5.3 Experimental Results

1. *Comparing cumulative energy consumption energy optimization algorithms:*

Figure 5.5 shows the cumulative energy consumption of different algorithms. As we can observe, activating sensors based on space-state demanded accuracy that has been considered in Context-Power-Aware algorithm reduces energy consumption by half compared to the case when all IoT devices in the location are running. On the other hand, the Node-approximation algorithm saves nearly 75% of energy, then, System-wide greedy algorithm consumes little less energy as it takes into consideration the best choice to activate across all devices.

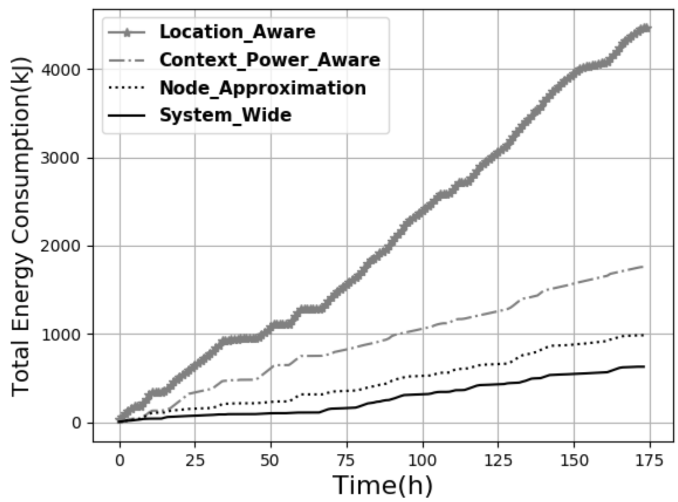


Figure 5.5: Cumulative energy consumption

2. *Total verses transmission energy consumption:*

Figure 5.6 shows the comparison among different algorithms in terms of the ratio of energy consumed for transmission compared to the total energy cost. The resulting measurements indicate that a significant component of the total energy cost is consumed for communication activities.

3. *Exploiting multiple interfaces :*

We next study the effect of using multiple network interfaces on the overall energy consumption. Figure 5.7 illustrates the energy consumed by the system-wide greedy

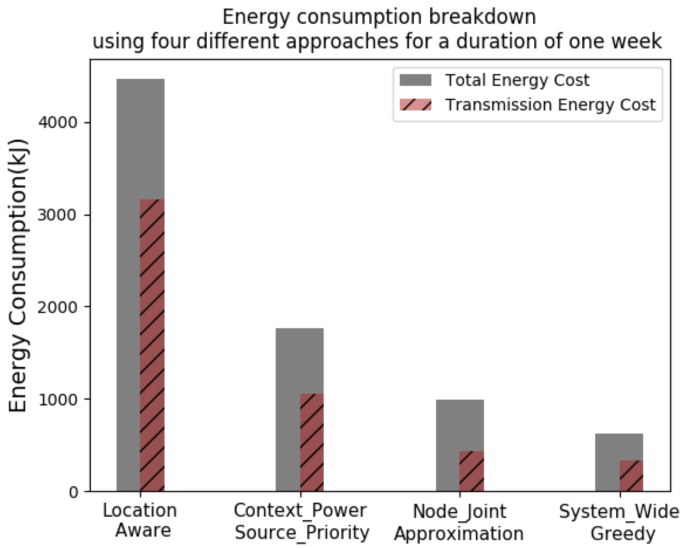


Figure 5.6: System-wide on multi-Network settings

algorithm under three settings (only 1 interface for each IoT node(Wi-Fi); from 2-4 interfaces; the extreme case where all nodes have 4 interfaces (Wi-Fi,Bluetooth,NB-IoT, LTE-M). As can be seen, increasing the connectivity options allows the System-wide greedy algorithm to reduce the energy dissipation over time.

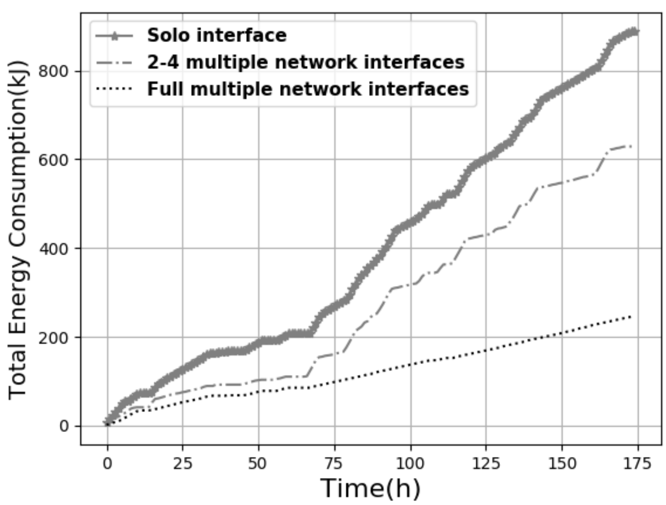
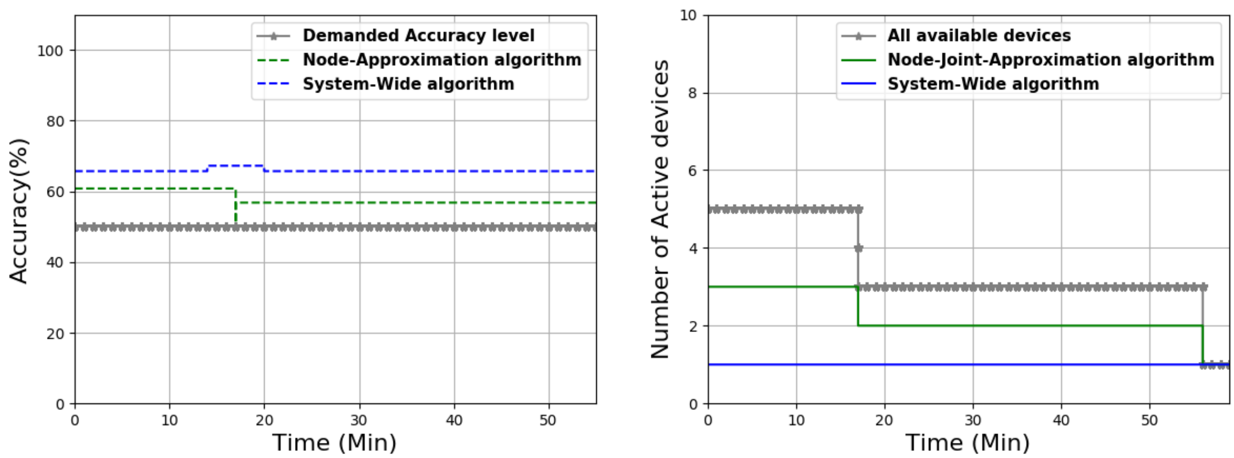


Figure 5.7: Transmission energy verses Total energy

4. *IoT Density and Scalability Studies:*

Given that the above results that indicate the efficacy of the proposed greedy techniques, we explore the scalability of our approach in the context of the two greedy algorithms - Node-approximation and System-wide algorithms. Specifically, we focused on a one-hour window, where we observe that both techniques deliver near demanded accuracy - Figure 5.8a. An interesting result can be seen in Figure 5.8b where the total number of available IoT nodes in the one-hour segment is five.



(a) Accuracy comparison

(b) Number of available/active devices

Figure 5.8: Comparison: Accuracy and Number of activated devices

The Node-approximation algorithm activates 3 out of 5 nodes to reach the demanded accuracy. In comparison, the System-wide algorithm activates only 1 out of the 5 nodes. An indirect consequence of using fewer nodes without sacrificing accuracy is that the System-wide algorithm can support longer operational lifetimes and hence increase reliable operation of the desired service.

In the next experiment, we studied the effect of node density. As seen in Figure 5.9, the System-wide greedy algorithm bounded the energy cost even with a dense IoT deployment, which is again beneficial in terms of system reliability and sustainability.

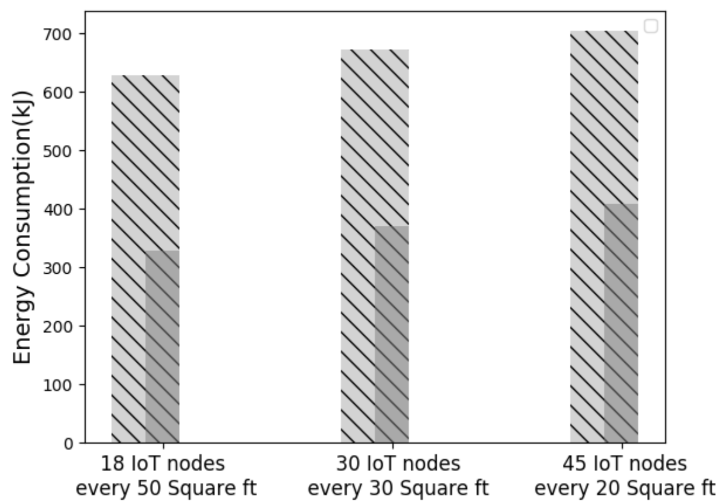


Figure 5.9: System-wide on Multi-scale settings

5.6 Chapter Summary and Discussion

In this chapter, we studied techniques to ensure energy efficiency at the network layer for perpetual IoT applications in smartspaces while ensuring application service quality. In particular, we considered how best to exploit the presence of multiple sensing modalities and multiple network interfaces along with knowledge of the application needs in the underlying space to intelligently activate the underlying system configuration.

In mission-critical environments (e.g. hospitals, chemical facilities, nuclear power plants), perpetual monitoring is critical to ensure safe operation; the extraction and exploitation of current operating context is critical for timely response. The notion of space-state based monitoring and activation in this work is a starting point to support both efficiency and safety in these settings. The ability for such cross-layer coordination (application, networking and devices) is of increasing importance as the number of IoT devices and connectivity choices increase - such flexibility also enables providers to expand on existing deployments as new technologies emerge.

Chapter 6

Computation-Aware Edge Resource Optimization

6.1 Chapter Overview

In chapters 4 and 5, we developed scheduling techniques to improve the efficiency for sensing and transmission tasks. However, efficient scaling of back-end IoT systems that serve multiple users/applications remains a challenge. In particular, enabling real-time computation to analyze insights obtained from sensory data is challenging in the context of perpetually operating IoT platforms. In this chapter, we discuss how to enable efficient computation in community-scale IoT systems. We expand our senior use-case setting from a smart home to a senior facility with multiple individuals. In this multi-user setting, we address the issue of computation level optimization.

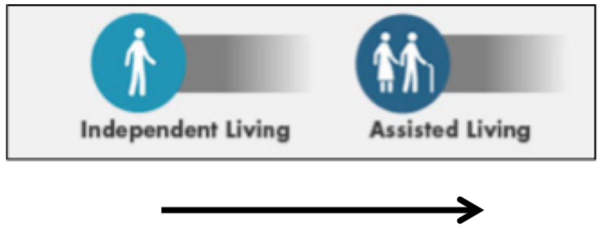


Figure 6.1: Scaling up the settings from independent living to an assisted living in a senior facility which adds new levels of computation complexity.

6.2 Edge Computing for scalable perpetual operations

Providing real-time services to monitor multiple individuals in an assisted-living facility is crucial, missing or delaying the detection of a critical event (e.g. a serious fall) can impact outcomes for the older adult. Typically, senior facilities are continuing care retirement communities (CCRC) with multiple levels of care. The needs of residents in different segments of a CCRC vary. The design goal is to enable seniors to stay in the same community with familiar surroundings and caregivers, even if their health conditions change down the road.

Figure 6.2: Continuing care retirement community plan example

A typical continuing care retirement community is illustrated in figure 6.2. Four levels of care are common here: Independent living, Assisted living, Memory Care and Nursing homes. Independent living is designed for individuals who can do most activities of daily living without assistance. While assisted living is designed for individuals who need assistance and services like medication management, housekeeping, etc, memory care provides help for individuals with memory issues. Nursing homes are designed to support individuals with high medical care needs for e.g. after an injury or a procedure.

These senior living communities strive to achive a balance between a safe environment and an independent one; this is challenging where needs are dynamic and vary on a day-by-day basis. Perpetual monitoring (using multimodal data 'sensors' and analytics) can help senior care providers balance resident independence and safety. Such monitoring is achieved by collecting a large volume/items of data in a relatively unobtrusive fashion using wearable technology and connected devices. Senior care communities are now enabled with adequate visibility and ability to monitor residents in less intrusive, more targeted ways based on residents' needs.

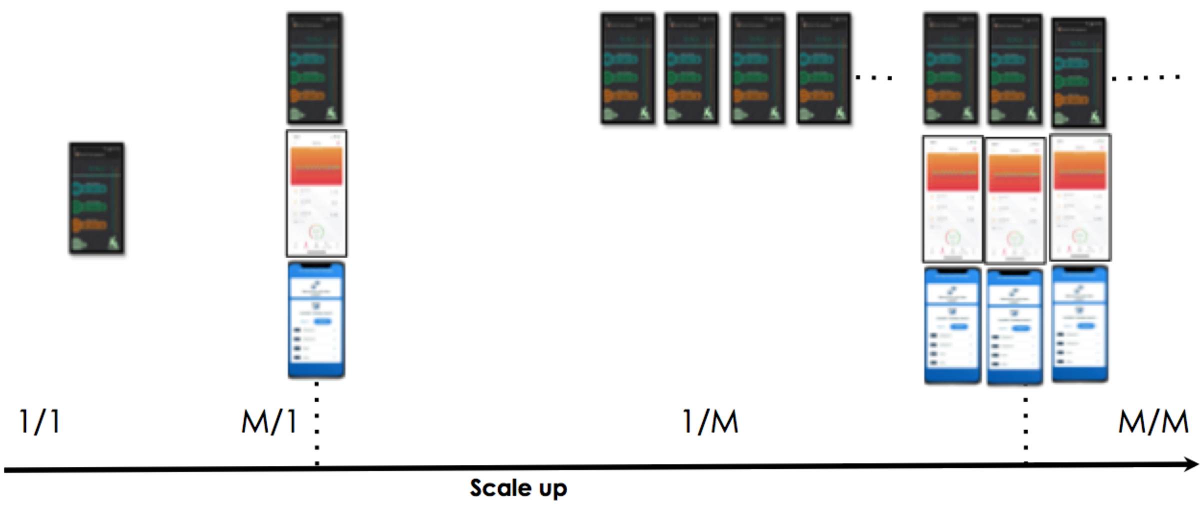


Figure 6.3: Many instances of different applications are running simultaneously

In order to ensure safety and independence for those individuals who live in these different levels of care communities, we need to run multiple monitoring applications. Examples of particular apps are: fall detection, gait monitoring, disease symptoms monitoring, medication management, etc. The multiple instances of different applications are running simultaneously (e.g., multiple patients in the same community and requires many analytics for each patient) so we scale it up from one to many as shown in figure 6.3.

As we stated before, these monitoring applications collect and process huge amounts of data. One critical question is where to process this data/information. Three options come to mind for processing levels/locations- on devices, edges or clouds. Processing data in devices is challenging for many reasons: IoT devices mostly are heterogeneous with limited capability in terms of storing and processing power resources as they often rely on limited battery capacity. Furthermore, some applications may require data to be fused from multiple devices, the question is how to distribute and manage the computation overheads.

While processing data on clouds imposes high network latency, a normal approach is to preform computation on cloud platforms where abundant resources are available which becomes a bottleneck for real time applications- such as assisted living where timely processing is a key requirement. Additionally, any contingency issue or cloud service failures can pose a challenge. For example, on Nov 25 2020, AWS was down for a relatively long time and the well-known iRobot and the associated applications were affected, as a result.

There are multiple drawbacks of analyzing on cloud platforms for computation of sensory data on our setting. An alternate approach is to provide support for compute and analytics close to where data is created. This reduces the amount of data that needs to be transmitted to remotely located cloud platforms. In recent years, edge computing technologies promise to make possible computing and storage services. IoT devices can connect through low energy connectivity options to local edge gateways. By offloading the computationally intensive workloads to edge servers, quality of service metrics e.g., network transmission delays and transmission energy consumption, can be improved greatly.

In this work, we aim to utilize edge resources to handle the computational complexity caused by scaling up perpetual operations. To run an edge system perpetually and efficiently, we plan to exploit users semantics as people come with diverse needs based on their health conditions, and dynamic workloads such as the knowledge of the activities of daily living (ADLs) to run several applications with different levels of complexity. Therefore, the amount/latency/quality of data needed to be captured/transmitted, and the complexity of algorithms needed to process this data are different to deliver services without loss of quality.

6.3 Focus applications

We will be considering in this study three day-to-day safety applications for older adults: fall detection, gait monitoring and medication management. *Fall detection* as we mentioned in previous chapters is a major challenge in the public health care domain - the Center for Disease Control reports over 2 million falls annually in the United States and one out of three older people fall each year. Falls are the leading cause of death in older Americans, and for those injured, the medical costs are expected to reach $67.7 billion by the end of 2020, according to the National Council on Aging.

Secondly, *gait monitoring* has recently become an increasing interest among researchers for observation of the human movement, especially walking, which can refer to physical data of the individuals, that can be a useful information for point-of-care preventive medicine. Also, a continuous gait monitoring as gait speed, cadence, stride length, and gait variability are well-known indicators of functional ability that can provide fall risk assessment and allow timely interventions aiming for preventing falls. Many wearable devices or smartphone with built-in accelerometers have been used as an innovative way to assess gait; the sensors are placed on several locations, such as the pelvis, wrist[27], ankles[43], bag[89], and pocket. Meanwhile, a number of smartphone applications are available for gait assessment, such as Lockhart Monitor, developed by biomedical engineer Thurmon Lockhart. Another approach to assist gait and stability is based on sensors that have been applied to each sole/foot/shoe, e.g. [75, 71, 32].

Lastly, *medication management* is an important service for seniors, according the Center for Disease Control, almost 40 percent of seniors take five or more prescription drugs daily over a 30-day period. Common characteristics found in medication management approaches include recorded dosing events, stored records of adherence, audio-visual reminders to cue dosing, digital displays, real-time monitoring and feedback on adherence performance, e.g. [93].

Those three applications are interrelated, some medications affect gait/stability of the person which may cause falls. Some medications are among the most common causes of increased fall risk in older people. Researchers have developed many approaches to collect data to deliver these services/applications, each with different level of accuracy and complexity. If we look at different fall detection approaches as an example, they use diverse sensors such as 3D-accelerometer sensors, microphone, cameras, etc. and produce multi-modal data that will be processed using an algorithm with a complexity level ranging from light conditional algorithms to heavy machine learning algorithms. Also, the number of participated sensors, the amounts of 'time-series' data produced by some sensors can range from a small amount of data to big data. Each approach delivers different accuracy, and none of them in any real life situation can deliver 100% accuracy. For this reason, sometimes we need to run a combination of approaches to increase the accuracy for critical application. Therefore, increasing data, time and complexity of algorithms are the price for higher accuracy.

Approach	Sensor Type	Sensed Data	Accuracy	Algorithm
Wearable	Accelerometer	Text(X,Y,Z)	63%	Simple Conditional
	Acc. on phone	Text(X,Y,Z)	70%	Conditional
Ambient	Wi-Fi	Channel State Information (CSI)	94%	Random Forest (RFA)
	SamrtPad	Pressure data	70%	Connected Components
	Doppler Radar	Radio Waves	N/A	Support Vector Machine (SVM)
Vision	Video Surveillance cameras	Video	90.5%	Shape Features
	Camera + Acoustic	Image + audio	85.0%	2D Convolutional Neural Network(NN)
	Web camera	Video	N/A	CNN

Table 6.1: Fall Detection Approaches Examples, Accuracy vs. Computation Complexity

6.4 Exploting Semantics: "Person-Space State" Abstraction

In this section, we introduce personal-space-state strategy, a unique abstraction for spaces/individuals contexts.

In any assisted living facility, residents are diverse in their needs based on their physical abilities, mental issues, risks and chronic diseases. And as we mentioned previously, the goal of these facilities is to balance between safety and independence of residents. Many assisted-living continuous monitoring applications can be leveraged to achieve that balance; however, the continuous monitoring services/applications for all residence have many challenges, one of them is processing. Therefore, we will propose a description of a set of feasible modeling policies along with a practical semantic strategy to deliver efficient services that are tailored to meet the needs of each individual with the lowest processing overhead.

First, we proposed some static policies:

- Modeling residents based on their location, as shown in Figure 6.2, senior facilities have multiple levels of care as residents' needs in these communities are different.
- Modeling residents based on their health profile (E-profile), currently most senior facilities use paper documents and colors wristbands to keep track of seniors needs based on their risks, such as fall, limb alert, allergy, etc., as shown in figure 6.4.

Figure 6.4: Color-coded Wristband (examples)

Instead, we will be creating an E-Profile for each resident. Based on each individual (age, needs, fall risk, medical history) a classification level of monitoring is assigned (e.g. A, B, C) and apply different policies for each class. Classification process, as shown in figure 6.5, starts with a collection of health data about each individual, then an assessment based on flowcharts from researchers or authorized agency, such as CDC, that classify individuals to classes, each receives diverse services with different level of quality.

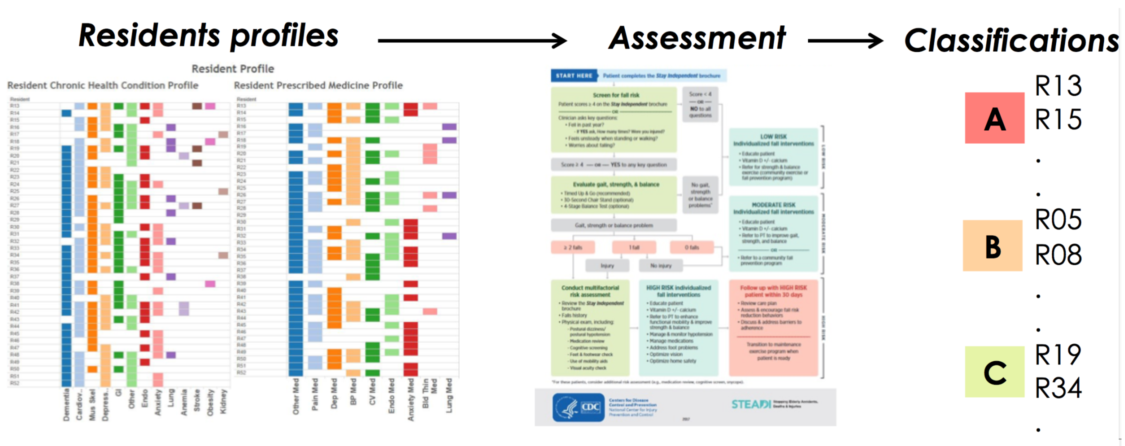


Figure 6.5: E-Profile residents classification

Secondly, we exploit dynamicity of workload, such as:

- *Health conditions* of individuals change down the road, such as heart disease, dementia, etc. the monitoring level should change accordingly.
- *After surgeries* usually individuals need more care and monitoring.
- *Medication* may incresses risks and needs, as researchers show that there is a pattern of dynamicity on seniors daily life, for example fall risks increase after use of some medications, as shown in figure 6.6.
- *Day/night* pattern change.
- *Emergency situations* some of them are predicted, and some are unpredicted.

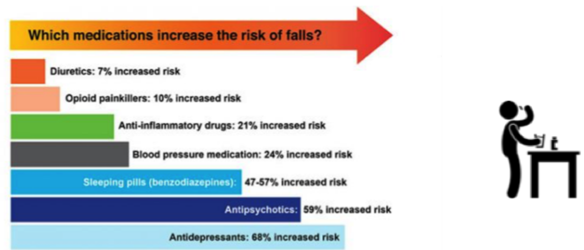


Figure 6.6: Example: some medication increases fall risks

Therefore, we introduce a key abstraction for exposing residence diversity and space/workload dynamicity in IoT deployments to build more reliable and energy-efficient processing systems. To capture dynamicity, we introduced earlier the concept of *Space-State* in our previous work[11], to optimize the selected subset of sensors that are sensing and transmitting for monitoring applications. Space-State states that under different contextual conditions, the amount/quality of data needed, accuracy of detection, differs; there are multiple states that the space shifts between them based on events occurring. In this work we took this concept further, one level more. So, there are multiple *Personal-Space-State* as shown in figure 6.7 that the space around each individual shifts between them based on his E-profile and events occurring(described by a state diagram), as shown in figure 6.8.

Figure 6.7: What is Personal-Space-State?

6.8

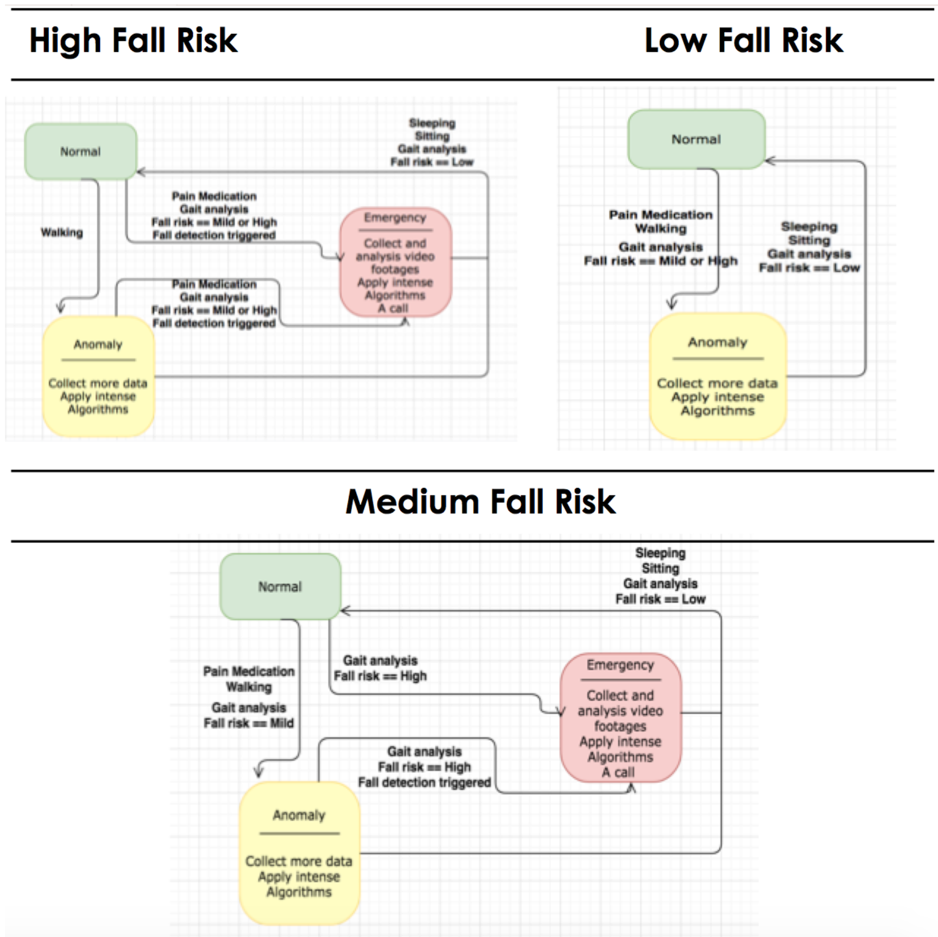


Figure 6.8: Transitions in state diagrams for three E-profile examples

6.5 In-Depth Characterization of Edge Resources Optimization Problem for IoT Services in Smart Spaces

In this section, we first introduce various terms comprising of our problem then discuss our assumptions and define frequently used terms and notations about the system; we formulate the problem as an NP-hard optimization problem.

6.5.1 Terms and assumptions

We assume that we have a deployed edge that can process all services for all senior-facility residences in its most accurate setting. Moreover, we assume that in each senior-facility there is a *Edge*, which is a fixed device that has the capability to process all services for all senior-facility residences in its most accurate setting and the ability to connect with all IoT devices through different protocols, such as Wi-Fi, Bluetooth, Ethernet, and ZigBee, etc. to track and collect IoT data streams.

6.5.2 Terms and Notations

Each Service can be achieved by diverse algorithms, each can operate in different *configurations* in terms of choosing values for their different parameters, such as in video algorithms we can choose the window intervals, sampling rate (fps) and computation frequency. The variation of these values results in different amount of computation consumption rate and varying degree of accuracy level across the different configurations of a particular algorithm. We define *accuracy* of an algorithm for a given configuration to be the probability of detecting the critical event when the algorithm operates in that configuration for a certain amount of time (referred to as the operation cycle). Obviously, there is a trade-off between the computation consumed by an algorithm at its different configurations and their accuracy levels. Higher accuracy is desired but only at the cost of higher computation cost, which leads to more resource and energy consumption. We are required to choose configurations for the algorithm so that edge operates efficiently.

6.5.3 Problem Statement and Formulation

We formulate EDGE RESOURCES OPTIMIZATION FOR IoT SERVICES PROBLEM as a constrained optimization problem as follows. We have S set of needed services, $s = 1, \cdots, S$. Each Service can be achieved by different algorithms. Let Service S has l_s algorithms $(1 \leq g \leq l_s)$.

Each algorithm can be described by a *profile*, which consists of different configurations the algorithm can run. Let algorithm g have l_{sg} configurations, and c_{sgk} and a_{sgk} denote the computation cost and accuracy level respectively for configuration k of algorithm g $(1 \leq k \leq l_{sg})$. Once a configuration is chosen for an algorithm, it operates for a certain amount of time before the next configuration is chosen. This duration, the operation cycle, is denoted by T.

We have a group of individuals, each person(individual) denoted as p, who need to use a set these services, each of which has its own *demand accuracy*, denoted by α_{sp}. The demand accuracy is the level of accuracy that all *actively running* algorithms should at least achieve. We argue that when multiple algorithms run to deliver a services, the combined accuracy increases. For example, if two algorithms independently detect a critical event with accuracy, which is the probability of detecting the critical event, equals to a_1 and a_2, then the combined accuracy will be the probability that at least one of them is detecting the event. That is:

$$\text{combined accuracy} = 1 - (1 - a_1)(1 - a_2)$$

The demand accuracy is a variable that changes depending on the personal-space-state. For example, the demand accuracy of fall detection service will be higher if the individual's personal-space-state is anomaly as he is a high fall-risk and just take a medication that affects his stability. The more the personal-space-state is crucial, the higher the demand accuracy should be.

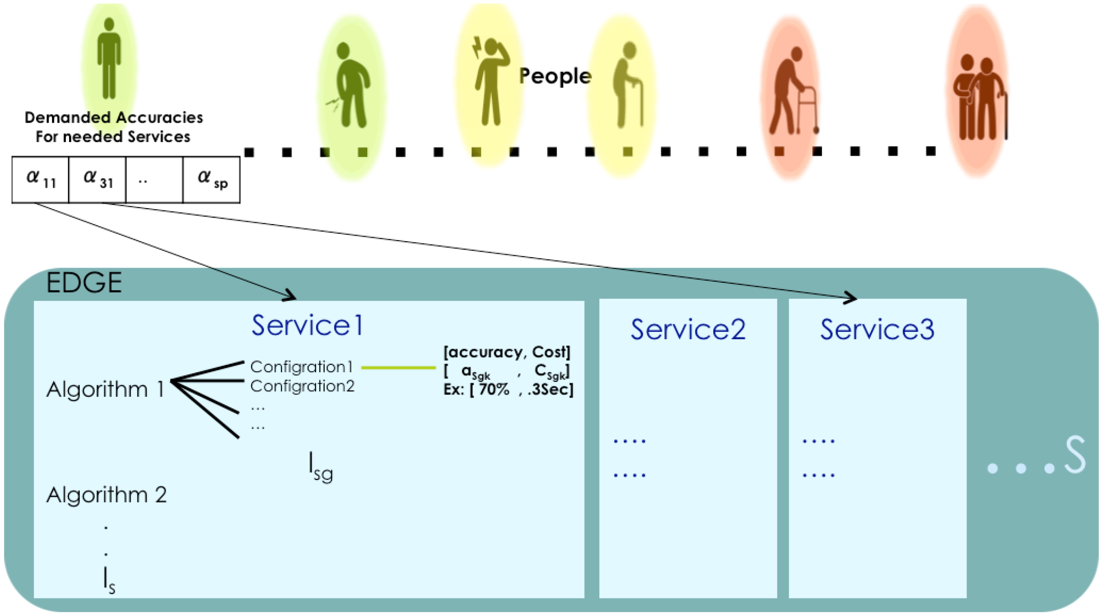


Figure 6.9: Edge resources optimization for IoT services formulation

Therefore, the demanded accuracy α_{sp} for each individual can be defined in many ways. It can be be based on static profile or dynamic semantics or both, which we considered in this paper.

We want to select the optimal algorithms subset with their appropriate configurations, so that the total computation cost remains as low as possible, while keeping the expected level of accuracy from the selected configurations above the demanded accuracy level α_{sp}.

Considering all these issues, we define a *cost function*, denoted by c_{sgk}, for each configuration of an algorithm as follows:

$$c_{sgk} = CPU_{time} \cdot T \tag{6.1}$$

The cost function captures the cost of running algorithm g to deliver service s in configuration

k. The cost is directly proportional to the amount of CPU consumed during the cycle.

We obtain an optimization formulation that chooses the configurations minimizing the overall cost of operation subject to the constraint that the combined accuracy level remains equal or above the demand accuracy. For the ease of exposition, we introduce an *idle* configuration (configuration 1) for each algorithm that has zero accuracy at zero or low cost. This allows all algorithms to be running exactly one configuration. By denoting x_{sgk} to be the binary variable indicating whether we choose configuration k of algorithm g to deliver service s, we have the following optimization problem:

$$\begin{aligned} \text{minimize} \quad & \sum_{s=1}^{S}\sum_{g=1}^{l_s}\sum_{k=1}^{l_{sg}} x_{sgk} \cdot c_{sgk} \\ \text{subject to} \quad & 1 - \prod_{g=1}^{l_s}\prod_{k=1}^{l_{sg}} (1 - x_{sgk} \cdot a_{sgk}) \geq \alpha_{sp} \\ & \sum_{k=1}^{l_{sg}} x_{sgk} = 1, \forall g \\ & \forall x_{sgk}, \in \{0,1\}, \forall s, g, k \end{aligned} \tag{6.2}$$

We can simplify constraint (6.2) as follows:

$$\begin{aligned} & 1 - \prod_{g=1}^{l_s}\prod_{k=1}^{l_{sg}} (1 - x_{sgk} \cdot a_{sgk}) \geq \alpha_{sp} \\ & \ln \prod_{g=1}^{l_s}\prod_{k=1}^{l_{sg}} (1 - x_{sgk} \cdot a_{sgk}) \leq \ln(1 - \alpha_{sp}) \\ & \sum_{g=1}^{l_s}\sum_{k=1}^{l_{sg}} x_{sgk} \cdot \ln(1 - a_{sgk}) \leq \ln(1 - \alpha_{sp}) \end{aligned}$$

Consequently, we obtain:

$$
\begin{aligned}
\text{minimize} \quad & \sum_{s=1}^{S} \sum_{g=1}^{l_s} \sum_{k=1}^{l_{sg}} x_{sgk} \cdot c_{sgk} \\
\text{subject to} \quad & \sum_{g=1}^{l_s} \sum_{k=1}^{l_{sg}} x_{sgk} \cdot \ln(1 - a_{sgk}) \leq \ln(1 - \alpha_{sp}), \forall s \\
& \sum_{k=1}^{l_{sg}} x_{sgk} = 1, \forall g \\
& \forall x_{sgk}, \in \{0, 1\}, \forall s, g, k
\end{aligned}
$$

The EDGE RESOURCES OPTIMIZATION FOR IoT SERVICES PROBLEM is an NP-hard problem that can be reduced from the Minimum Multiple Choice Knapsack Problem. The knapsack problem is known to be a well-studied NP-hard problem and a special case of the multiple choice knapsack problem with the feature that each item is in a group of its own [53].

In the MINIMUM MULTIPLE CHOICE KNAPSACK PROBLEM there is a set of items which are partitioned into groups and each item has a benefit and a weight. The objective of the MMKP is to find the least profitable set of items such that the total weight of the selected items is at least the weight limit [48].

Similarly, in the EDGE RESOURCES OPTIMIZATION FOR IoT SERVICES PROBLEM, the goal is to select a set of algorithms that minimize the total cost with exceeding the demanded accuracy threshold. Each algorithm has a set of configurations, including the option of not selecting it. Therefore, each algorithm defines a class from which we are selecting at most one option.

6.6 Description of Algorithms and Heuristics

Due to the NP-hardness of our problem, we use heuristics to help with the optimization, in this section, we will propose some naive algorithms followed by a family of feasible techniques to address the problem.

6.6.1 Base-line approaches

Developing an optimal IoT processing system for perpetual operation needs comprehensive knowledge about the semantics patterns, residences status and algorithms/services profiles. To handle the complexity that arises due to the dynamic nature and the diversity of the underlying users/infrastructure/services, we propose the following cross-layer system architecture as illustrated in Figure 6.10.

- **Complete algorithm**: one naive opportunistic approach is the *Complete algorithm* that runs the most accurate configuration for all algorithms/services/residences as long we have an adequate processing capacity. Obviously, that would deplete resources/energy from the edge without raising system accuracy/efficiency much.
- **Simple inefficient algorithm**: since our goal is to reduce the processing overhead, in this approach we activate all algorithms on the lowest configuration. Obviously, that would reduce the overhead in the edge but it reduce the efficiency of the services as it increases the missing critical events.

6.6.2 Context-Aware approaches

Services should be tailored to meet the needs of each individual is the notion behind these approaches. These algorithms internally take the data from the available resident health

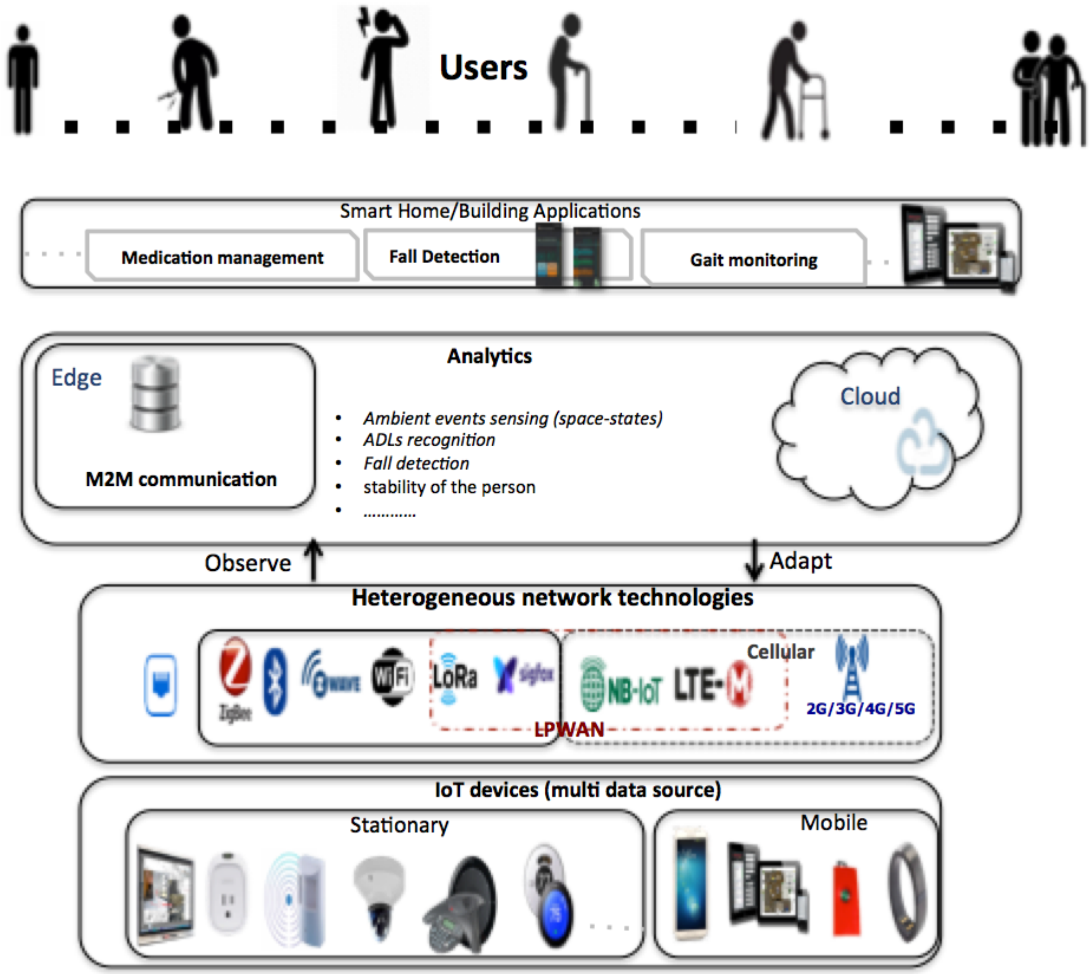


Figure 6.10: System Architecture

information databases and the different ambient light-weight sensors as input and provide classified and abstracted actionable insights as an output to the application/services layer.

- **E-profile algorithm (Static policies)**: this algorithm is utilizing the residents diversities in ages, needs, physical abilities, mental issue, risks (such as falls), chronic diseases, etc. Theses data/information are collected, documented and updated for every individual frequently in any facility. Services should be tailored to meet the needs of each individual. Therefore, based on the these profile databases, we can classify residents on different categories to deliver different levels of monitoring services.

- **Generalized space-state algorithm (Dynamic policies)**: in this approach, we take into the consideration the environment dynamicity that abstracted to space-states **(normal, anomaly, emergency)** this notion was introduced earlier in our previous work [12]. Each space-states mode requires different level of the demanded accuracy of operation. In this algorithm, we applied for each individual a generalized model on events and activities(ADLs) that span multiple environment settings and resident types.

- **Personal- Space-State Transition (PSST)**: in this approach, we combine the static and dynamic policies in a Two-phase algorithm. The first phase starting with initializing the set of needed services for each individual based on his recorded profile and current activity. Then, we calculate current personal-space-state *demand accuracy*, denoted by α_{sp} for each individual/needed service.

  In the second phase, for each service we rank the configurations per algorithm based on accuracy in an ascending order. Then, once the configurations for all algorithms are ranked, the heuristic builds the solution as follows:

  - Starting with an empty list then calculate *Utility metric* for all remaining algorithms. $Utility(i) = \frac{\Delta Accuracy}{\Delta Computational-cost}$.
  - the goal is to move toward the algorithm that offers lower change in *cost* compared to a big change in *accuracy.*
  - Choose the algorithm with the largest *Utility*, and calculate combined accuracy for all activated algorithms.
  - Keep repeat upgrade the chosen algorithm to the next available accuracy; the iteration ends when the accuracy demand is met.

  The algorithm mainly is progressively making a sequence of changes in which the current choices of algorithms/configurations are upgraded to the next best based on the gradient of accuracy change to cost change.

6.7 Performance Evaluation and Results

In this section, we describe the dataset, simulation environment and experimental design for performance evaluation, and present the results with analyses.

6.7.1 Dataset and Initial Measurements for performance evaluation for the Fall Detection System

To understand the actual cost, we started to look for fall detection video processing data-sets as One of the most complex approaches in computation is video processing, algorithms and measure computation cost with accuracy that can be delivered.

We were able to find a dataset[73] which is from University of Nice Sophia Antipolis that specifically looked at how complex it was to compute fall detection using video and its levels of related accuracies. So we utilized their findings to drive our results which will be confirmed with our measurements.

Algorithm BackGround Subtraction/Template Matching (BGSTM) which consists of 4 modules: Object Segmentation, Object Enhancement, Object Feature Extraction and Recognition events with an automatic alarm whenever FALL is occurred.

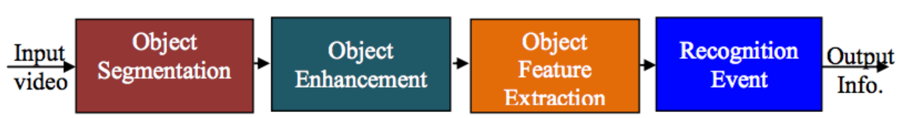


Figure 6.11: Block diagram of Fall Detection System[73]

Dataset A DUT-HBU (Danang University of Technology- Human Behavior Understanding) database which is classified with different actions: fall, non-fall in three camera directions is used to evaluate the efficiency of this system. It includes 42 fall videos and the rest is 59 videos. In this database, the fall direction is subdivided into three basic directions:

- Direct: object falls the same orientation with the direction of the camera.
- Cross: objects created the 30o -60o angles with camera when the falling occurs.
- Side: object falls in a perpendicular direction to the camera.

Figure 6.12: The position of falling compared with angles of camera[73]

Platform The video application was implemented on processor cores (ARM Cortex A9 processor) of ZYNQ Platform different configurations. TI USB Interface Adapter PMBus associated with TI Fusion Digital Power Designer GUI are used to measure the power consumption on processor cores. Zynq7000 AP SoC platform which has both processor cores and FPGA.

Execution time Figure 6.13 illustrates the comparison execution time at two image resolutions, 320x240 and 680x360, processing on one processor of Zynq 7000 AP SoC platform.

Accuracy The relationship between accuracy performances in the system with the other parameters such as frame rate, resolution, etc is crucial.According to [73], based on some experiments on various frame rates as presented in Table6.14, shows that the relationship between fps and accuracy is direct proportion.

Energy per frame An example of measured power/energy of fall detection system at two image resolutions 320x240 pixels and 680x360 pixels. The energy per frame with image

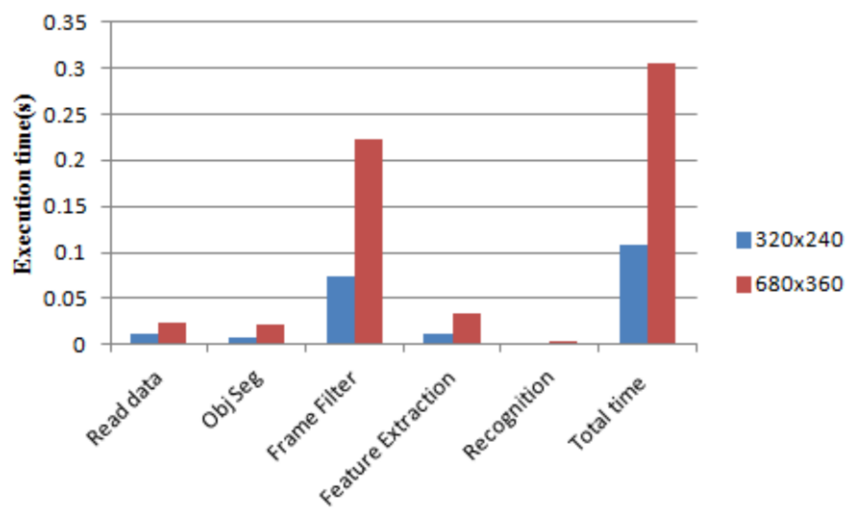


Figure 6.13: Comparison execution times at two image resolutions on one core [73]

Resolution	fps	Accuracy (%)
320x240	10	66.7
320x240	20	72.2
320x240	30	85.1

Figure 6.14: The relationship between Accuracy performance with frame rate of input video[73]

resolution (320x240) is 45.11 mJ, while it is 129.13 mJ for (680x360).

6.7.2 Experimental Setup - Simulation Studies

We applied the initial performance measurements for the fall detection system dataset that illustrated in the previous section[73], in a congregate senior living facility where there are 300 plus residents. Therefore, to monitor those 300+ residents we need to instrument about 300 cameras in private spaces and common spaces. A virtual site survey of a congregate living facility space, shown in Figure 6.15 was conducted and instrumented with these type of sensors to understand what is the cost of the processing.

To conduct further experiments, we developed a discrete-event simulator and created a test

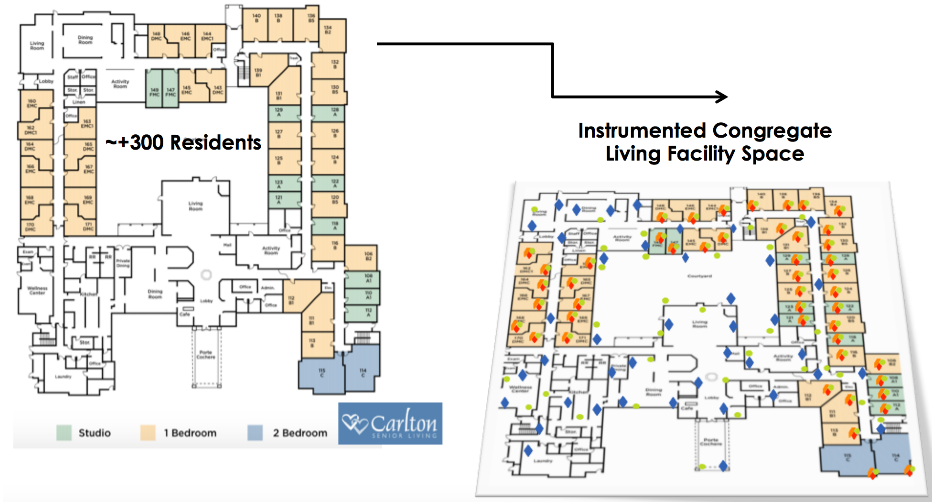


Figure 6.15: Virtual Site Survey for Congregate Living Facility Space (Example)

case based on real-world elderly living. We execute our simulations on activity recognition patterns for week ($\approx$ 170 hrs) of data from the eleven datasets the CASAS trace dataset obtained from [84] that contains the activities of daily living of twenty individuals (eleven CASAS datasets collected in seven environments, as shown in Figure 6.16) in an assisted living setting. A majority of the datasets recognize and track activities of daily living (ADLs) that people normally perform as part of their daily routines, such as: personal hygiene, sleep, bed-to-toilet, eat, cook, work, leave Home,enter Home, relax, take medicine and bathing. They recognized these ADLs from different light-wight ambient types of sensor: motion, door, or other; based on a generalized model that can be learned for common activities that span multiple environment settings and resident types.

According to CDC, every second a fall incident in the US for about 54 million senior population. Based on this statistical data, the incidents that can happen in one week for 300 people is about 3 falls. $(300*60*60*24*7)/(54*10^6) = 3.36$

Performance Evaluation Metrics:

- *Cumulative processing energy consumption*:

  The total processing energy consumption can be used as a benchmark to evaluate the

Figure 6.16: Sensor layout for the seven CASAS smart environment testbeds[84]

energy optimization algorithms. Processing energy cost represents a small portion of the whole IoT system energy consumption. However, we found that if we reduce the processing energy cost, we reduce the underlying costs (transmission and sensing). As an example, in our comparison of fall detection approaches, table 6.14, one of the most complex approaches in computation is video processing, which is also costly in transmission and sensing.

- Number of missing events: intuitively, an optimized algorithm should increases the reliability and reduce the number of missing events while attaining the demanded accuracy.

  E(Missing event) = (total events)* (1- accuracy)

  (total events)= total time * (number of event occurs per hour)

6.7.3 Experimental Results

Total energy consumption comparison:

Figure 6.17 shows the total energy consumption in different algorithms. As we can observe, processing the most available accurate sensed data in our data set[73] (30 frame per second) which deliver one service (fall detection service) for all residents (300+ senior) consume huge amounts of energy, which shows that if we consider more accurate data sensing, run more services for each individual and scale-up to cover more residents, it raises energy cost dramatically. The other impractical algorithm is the simple one, which run the least accurate configuration for all algorithms/services/residences which obviously will decrease the energy cost along the reliability/QoS of the system.

Then, we show the three context algorithms: *E-profile, Generalized space state, and Personal space state*. Clearly, they reduce the energy cost without sacrificing the service quality, by considering spaces state and personal profiles, as they have a direct association to the level of care needed by every individual at any circumstance.

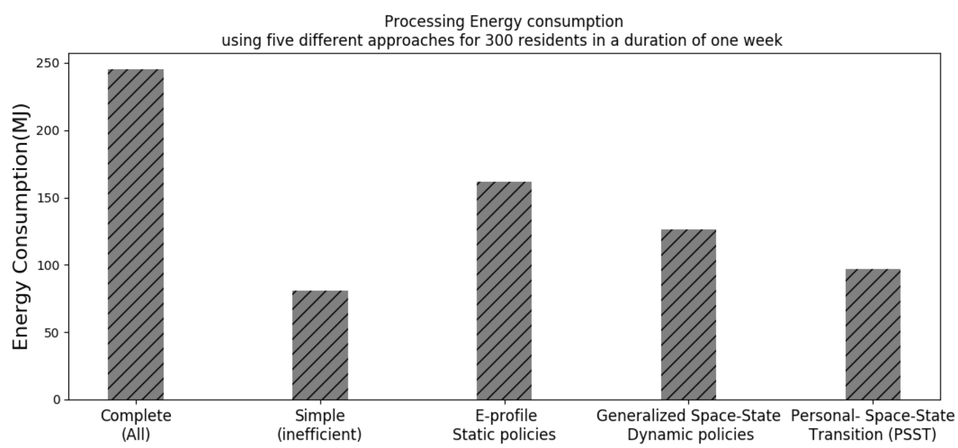


Figure 6.17: Approaches Comparison- Energy Consumption

Reliability vs. energy consumption comparison:

To understand the accuracy of our proposed approach, we measured missed events (falls) using different algorithms. Note that the probability that a fall event will be missed lies between 1 and 0 (in our experimental data set with 3.4 falls in the group of 300 residents per week). In our experimental results, the complete and personal-space-sate-algorithm were similar with miss probabilities of almost zero. On the other hand, the simple, E-profile and Generalized space-state approaches reported fall events as being missed. Overall, the PSST technique showed improved benefits along two angles — the ability to detect critical events accurately while reducing the overall energy consumption and computational overhead on the system side.

6.8 Chapter Summary and Future Work Discussion

In general, vision based approaches provide better sensing with higher levels of application QoS, However, they have been prohibitive to deploy at scale due to computational complexity of vision processing and deployment cost. In recent years, the cost of visual sensing is coming down due to low camera prices and can potentially be used to support monitoring at scale.

In this work, we tested our approach and algorithms based on UNS dataset [73] that offers energy consumption breakdown for processing a vision fall detection. We are continuing exploring the energy cost of vision-based processing, (measured in CPU cycles) in the context of fall detection applications. For this, we are collecting measurments based on URFD dataset, apply CNN-based ML techniques and optical flow methods to detect falls from surveillance cameras.

Chapter 7

Conclusion

The Internet of Things (IoT) is becoming a critical component in the realization of perpetual awareness and mission-critical systems for smart homes and communities. The combination of space-context, human-oriented semantics and physical IoT deployments is a promising contribution towards this goal. In this thesis, we demonstrate that inserting the human and semantic aspects as an integral part of how perpetual systems operate and are designed have a beneficial impact on sustainability and efficiency.

In this thesis, the initial efforts have been devoted to deploying (SAFER), a perpetually operating personal sensing smarthome platform, building on the successes of the (SCALE Project) at a senior living facility in Montgomery County, MD [17, 1, 97]. We conducted measurement studies and identify major IoT perpetual monitoring challenges. To provide focus to our work, we considered elderly fall detection as as driving usecase. Falls among seniors are incredibly common, and can cause serious injuries. Each year, three million older adults go to the ER for injuries due to falls. Heterogeneous approaches to enable Fall Detection in real-time are studied and implemented to extract the measurements for event detection accuracy and power consumption.

In Chapter 3, we proposed novel cross-level context-aware methodology that utilizes semantics, heterogeneity and real-time scheduling to define the low cost architecture for which presents a sufficient accuracy rate. Our optimization techniques are across three layers: sensing, communication, and processing. We believe that such techniques are essential to creating deployable IoT for mission-critical societal applications that require perpetual operations such as healthcare and assisted living.

In Chapter 4, we started by leveraging the concept of activities of daily living (ADLs) for energy-optimized sensor activation to create, a perpetual IoT awareness platform.

In Chapter 5, we considered how best to exploit the presence of multiple sensing modalities and multiple network interfaces through the use of dynamic Space-states and knowledge of the application needs in the underlying space to intelligently activate the underlying system configuration.

In Chapter 6, We discuss how to enable efficient computation in community-scale IoT systems by expanding our approach to include processing level and scale-up our setting with multiple individuals to address the issue of computation.

More broadly, we aim to enable and ensure the multiple functional and non-functional needs of societal scale applications by leveraging new emerging technologies - this will require an in-depth understanding of how these requirements interact. The ability for such cross-layer coordination (application, processing, networking and devices) is of increasing importance as the number of IoT devices, connectivity, processing choices increase - such flexibility also enables providers to expand on existing deployments as new technologies emerge. In future work, we will scale-up this approach by considering a wider range of scenarios, including intruder detection applications and fire safety applications with multiple people in the space, multi-service resource provisioning.

7.0.1 Additional Use cases

Our context-aware approach can be replicated and applied to any heterogeneous IoT deployment setting(e.g. safety, healthcare, assisted living and mission-critical) where constant monitoring is required. Theses deployments aim to improve the quality of life for those who are vulnerable, while helping them to lead as an independent lifestyle.

On another application scenario e.g. 'Intruder detection system' which is a network of integrated IoT devices working together with a central controller to protect against burglars and other potential home intruders. Our three-phase approach starts with a learning phase that captures the deployment setting by computing a floor-plan segmentation of the space as well as the participating IoT device profiles, including status and different configurations that are given infrastructure information. Then, In the second phase we perform activity recognition using our sliding window activity recognition based on different types of sensors such as smartphones, wearable sensors, cameras, and occupancy sensors. After this, based on the recognized location and activities' patterns the dynamic configuration phase adjusted the status of participating IoT devices in entry points, like doors and easily accessible windows, as well as interior spaces containing valuables like art, computers, etc. Then, it computes the optimal overall energy configuration by reducing the energy consumption on the attended areas, for example.

Bibliography

[1] Scale project. `https://scale.ics.uci.edu/`.

[2] Smart america challenge. `https://smartamerica.org/`. online.

[3] Tippers project. `http://tippers.ics.uci.edu/`.

[4] Product data sheet. Cisco Aironet 802.11a/b/g Wireless PCI Adapter. Technical report, Cisco Aironet, January 2010.

[5] Cellular system support for ultra-low complexity and low throughput Internet of Things (CIoT). Technical Report 11, 3GPP, TR 45.820, 2015.

[6] Evaluation of LTE-M towards 5G IoT requirements for Category-M1 Devices . Technical report, December 2017.

[7] B. Afzal, S. A. Alvi, G. A. Shah, and W. Mahmood. Energy efficient context aware traffic scheduling for IoT applications. *Ad Hoc Networks*, 62:101–115, 2017.

[8] Z. T. Al-Azez et al. Virtualization framework for energy efficient iot networks. In *CloudNet*, pages 74–77, 2015.

[9] B. R. Al-Kaseem, H. S. Al-Raweshidy, Y. Al-Dunainawi, and K. Banitsas. A new intelligent approach for optimizing 6lowpan mac layer parameters. *IEEE Access*, 5:16229–16240, 2017.

[10] N. Alhassoun, M. Y. S. Uddin, and N. Venkatasubramanian. Safer: An iot-based perpetual safe community awareness and alerting network. In *IGSC*, pages 1–8, Oct 2017.

[11] N. Alhassoun, M. Y. S. Uddin, and N. Venkatasubramanian. Context-aware energy optimization for perpetual iot-based safe communities. *SUSCOM*, 22:96–106, 2019.

[12] N. Alhassoun, M. Y. S. Uddin, and N. Venkatasubramanian. Multi-network provisioning for perpetual operations in iot-enabled smart spaces. In *2020 IEEE International Conference on Smart Computing (SMARTCOMP)*, pages 89–97, 2020.

[13] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Netw.*, 7(3):537–568, May 2009.

[14] R. A. Antonio, R. M. de la Costa, A. R. Ison, W. K. Lim, R. A. Pajado, D. B. Roque, R. Yutuc, C. V. Densing, M. T. de Leon, M. Rosales, and L. Alarcon. Implementation of dynamic voltage frequency scaling on a processor for wireless sensing applications. In *TENCON 2017 - 2017 IEEE Region 10 Conference*, pages 2955–2960, Nov 2017.

[15] F. Aoudia et al. Leveraging Energy Harvesting and Wake-Up Receivers for Long-Term Wireless Sensor Networks. *Sensors*, 18:1578, 2018.

[16] R. Balani. Energy consumption analysis for bluetooth, wifi and cellular networks . Technical report.

[17] K. Benson, C. Fracchia, G. Wang, Q. Zhu, S. Almomen, J. Cohn, L. D'arcy, D. Hoffman, M. Makai, J. Stamatakis, and N. Venkatasubramanian. Scale: Safe community awareness and alerting leveraging the internet of things. *IEEE Communications Magazine*, 53(12):27–34, 2015.

[18] K. E. Benson, Q. Han, K. Kim, P. Nguyen, and N. Venkatasubramanian. Resilient overlays for iot-based community infrastructure communications. In *2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pages 152–163, 2016.

[19] K. E. Benson and N. Venkatasubramanian. Improving sensor data delivery during disaster scenarios with resilient overlay networks. In *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pages 547–552, 2013.

[20] K. E. Benson, G. Wang, N. Venkatasubramanian, and Y.-J. Kim. Ride: A resilient iot data exchange middleware leveraging sdn and edge cloud resources. In *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pages 72–83, 2018.

[21] A. Biason, C. Pielli, M. Rossi, A. Zanella, D. Zordan, M. Kelly, and M. Zorzi. Eccentric: An energy- and context-centric perspective on iot systems and protocol design. *IEEE Access*, 5:6894–6908, 2017.

[22] R. Bormann et al. Room segmentation: Survey, implementation, and analysis. In *IEEE ICRA*, pages 1019–1026, 2016.

[23] A. Bourke, J. O'Brien, and G. ÓLaighin. Evaluation of a threshold-based tri-axial accelerometer. *J Gait and Posture*, 26:194–199, 01 2007.

[24] S. Brienza, D. D. Guglielmo, G. Anastasi, M. Conti, and V. Neri. Strategies for optimal mac parameter setting in ieee 802.15.4 wireless sensor networks: A performance comparison. In *2013 IEEE Symposium on Computers and Communications (ISCC)*, pages 000898–000903, July 2013.

[25] S. Chiaravalloti, F. Idzikowski, and Ł. Budzisz. Power consumption of WLAN network elements, 2011.

[26] C. Choi, J. Jeong, I. Lee, and W. Park. Lora based renewable energy monitoring system with open iot platform. In *2018 International Conference on Electronics, Information, and Communication (ICEIC)*, 2018.

[27] G. Cola, M. Avvenuti, F. Musso, and A. Vecchio. Personalized gait detection using a wrist-worn accelerometer. In *2017 IEEE 14th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, pages 173–177, 2017.

[28] D. Cook. Learning setting-generalized activity models for smart spaces. *IEEE Intelligent Systems, 2011*, 2011.

[29] D. J. Cook. Learning Setting-Generalized Activity Models for Smart Spaces. *IEEE intelligent systems*, 2010(99):1, sep 2010.

[30] A. S. Crandall et al. CASAS: A smart home in a box. *Computer*, 46:62–69, 2013.

[31] R. Du, L. Gkatzikis, C. Fischione, and M. Xiao. On maximizing sensor network lifetime by energy balancing. *IEEE Transactions on Control of Network Systems*, 5(3):1206–1218, Sept 2018.

[32] B. M. Eskofier, S. I. Lee, M. Baron, A. Simon, C. F. Martindale, H. Gaßner, and J. Klucken. An Overview of Smart Shoes in the Internet of Health Things: Gait and Mobility Assessment in Health Promotion and Disease Monitoring. *Applied Sciences*, 7(10), 2017.

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

[34] A. B. G. Smaragdakis, I. Matta. Sep: A stable election protocol for clustered heterogeneous wireless sensor networks. In *Second International Workshop on Sensor and Actor Network Protocols and Applications (SANPA)*, 2004.

[35] D. Ganesan et al. Networking issues in wireless sensor networks. *J. Parallel Distrib. Comput.*, 64(7):799–814, 2004.

[36] M. Gorlatova et al. Movers and shakers: Kinetic energy harvesting for the internet of things. *IEEE Journal on Selected Areas in Communications*, 33(8):1624–1639, 2015.

[37] C. Gray, R. Ayre, K. Hinton, and R. S. Tucker. Power consumption of iot access network technologies. In *2015 IEEE International Conference on Communication Workshop (ICCW)*, pages 2818–2823, 2015.

[38] C. A. Gray. *Energy consumption of Internet of Things applications and services*. PhD thesis, may 2019.

[39] Q. Han, R. Eguchi, S. Mehrotra, and N. Venkatasubramanian. Enabling state estimation for fault identification in water distribution systems under large disasters. In *2018 IEEE 37th Symposium on Reliable Distributed Systems (SRDS)*, pages 161–170, 2018.

[40] Q. Han, P. Nguyen, R. Eguchi, K. Hsu, and N. Venkatasubramanian. Toward an integrated approach to localizing failures in community water networks. pages 1250–1260, 06 2017.

[41] J. He, L. Duan, F. Hou, P. Cheng, and J. Chen. Multiperiod scheduling for wireless sensor networks: A distributed consensus approach. *IEEE Transactions on Signal Processing*, 63(7):1651–1663, April 2015.

[42] W. R. Heinzelman et al. Energy-efficient communication protocol for wireless microsensor networks. In *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*. IEEE, 2000.

[43] J. Howcroft, J. Kofman, E. D. Lemaire, and W. E. McIlroy. Analysis of dual-task elderly gait in fallers and non-fallers using wearable sensors. *Journal of Biomechanics*, 49(7):992–1001, 2016.

[44] Z. Huang, K.-J. Lin, S.-Y. Yu, and J. Y.-j. Hsu. Co-locating services in IoT systems to minimize the communication energy cost. *Journal of Innovation in Digital Ecosystems*, 1(1):47–57, 2014.

[45] E. Ibarra, A. Antonopoulos, E. Kartsakli, J. J. P. C. Rodrigues, and C. Verikoukis. Qos-aware energy management in body sensor nodes powered by human energy harvesting. *IEEE Sensors Journal*, 16(2):542–549, Jan 2016.

[46] R. Igual, C. Medrano, and I. Plaza. Challenges, issues and trends in fall detection systems. *BioMedical Engineering*, 12(1):66, 2013.

[47] M. A. Imran, A. Zoha, L. Zhang, and Q. H. Abbasi. Grand Challenges in IoT and Sensor Networks. *Frontiers in Communications and Networks*, 1:7, 2020.

[48] M. T. Islam et al. *Approximation algorithms for minimum knapsack problem.* PhD thesis, Lethbridge, Alta.: University of Lethbridge, 2009.

[49] D. Jung, R. Kim, and H. Lim. Power-saving strategy for balancing energy and delay performance in WLANs. *Computer Communications*, 50:3–9, 2014.

[50] P. Kamalinejad et al. Wireless energy harvesting for the internet of things. *IEEE Communications Magazine*, 53(6):102–108, June 2015.

[51] R. M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, 1972.

[52] O. Kazeem, O. Akintade, and L. Kehinde. Comparative Study of Communication Interfaces for Sensors and Actuators in the Cloud of Internet of Things. *International Journal of Internet of Things*, 6:9–13, 2017.

[53] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack problems.* Springer, 2004.

[54] M. E. Khanouche, Y. Amirat, A. Chibani, M. Kerkar, and A. Yachir. Energy-centered and QoS-Aware services selection for internet of things. *IEEE Transactions on Automation Science and Engineering*, 13(3):1256–1269, July 2016.

[55] E. Kim, S. Helal, and D. Cook. Human activity recognition and pattern discovery. *IEEE Pervasive Computing*, 9(1):48–53, Jan 2010.

[56] N. C. Krishnan and D. J. Cook. Activity recognition on streaming sensor data. *Pervasive and Mobile Computing*, 10, Part B:138–154, 2014.

[57] D. Kumar, T. C. Aseri, and R. Patel. EEHC: Energy efficient heterogeneous clustered scheme for wireless sensor networks. *Comput. Commun.*, 32(4):662–667, Mar. 2009.

[58] M. Lauridsen, R. Krigslund, M. Rohr, and G. Madueno. An empirical nb-iot power consumption model for battery lifetime estimation. In *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, pages 1–5, 2018.

[59] H. Li. An energy efficient routing algorithm for heterogeneous wireless sensor networks. In *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, volume 3, pages V3–612–V3–616, Oct 2010.

[60] H. Li and B. Sikdar. Optimal parameter selection for discrete-time throughput-optimal mac protocols. In *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1526–1531, April 2014.

[61] S. Li and J. Huang. Energy efficient resource management and task scheduling for iot services in edge computing paradigm. In *ISPA/IUCC*, pages 846–851, Dec 2017.

[62] Y. Li and Y. Liang. Temporal lossless and lossy compression in wireless sensor networks. *ACM Trans. Sen. Netw.*, 12(4):37:1–37:35, Oct. 2016.

[63] C. H. Liu, J. Fan, J. W. Branch, and K. K. Leung. Toward QoI and energy-efficiency in internet-of-things sensory environments. *IEEE Transactions on Emerging Topics in Computing*, 2(4):473–487, Dec 2014.

[64] C.-L. Liu, C.-H. Lee, and P.-M. Lin. A fall detection system using k-nearest neighbor classifier. *Expert Systems with Applications*, 37(10):7174 – 7181, 2010.

[65] Y. Liu, A. Liu, Y. Hu, Z. Li, Y. J. Choi, H. Sekiya, and J. Li. Ffsc: An energy efficiency communications approach for delay minimizing in internet of things. *IEEE Access*, 4:3775–3793, 2016.

[66] K. Machado, D. Rosário, E. Cerqueira, A. A. F. Loureiro, A. Neto, and J. N. de Souza. A Routing Protocol Based on Energy and Link Quality for Internet of Things Applications. *Sensors (Basel, Switzerland)*, 13(2):1942–1964, feb 2013.

[67] A. Mainwaring et al. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, WSNA '02, pages 88–97, New York, NY, USA, 2002. ACM.

[68] E. . C. F. . M. F. Mekki, Kais Bajic. A comparative study of LPWAN technologies for large-scale IoT deployment. *ICT Express*, 2019.

[69] S. Mittal and Z. Zhang. EnCache: Improving Cache Energy Efficiency Using A Software-Controlled Profiling Cache. In *IEEE International Conference On Electro/Information Technology*, Indianapolis, USA, may 2012. IEEE, IEEE.

[70] S. Mittal, Z. Zhang, and Y. Cao. Cashier: A cache energy saving technique for qos systems. In *2013 26th International Conference on VLSI Design and 2013 12th International Conference on Embedded Systems*, pages 43–48, Jan 2013.

[71] Y. Mittra and V. Rustagi. Classification of subjects with parkinson's disease using gait data analysis. In *2018 International Conference on Automation and Computational Engineering (ICACE)*, pages 84–89, 2018.

[72] M. Mubashir, L. Shao, and L. Seed. A survey on fall detection: Principles and approaches. *Neurocomputing*, 100:144–152, 2013.

[73] T. K. H. Nguyen. Low power architecture for fall detection system. *Université Nice Sophia Antipolis*, 2015.

[74] V. Nguyen et al. A scalable approach for dynamic evacuation routing in large smart buildings. In *SmartComp*, 2019.

[75] T. Nilpanapan and T. Kerdcharoen. Social data shoes for gait monitoring of elderly people in smart home. In *2016 9th Biomedical Engineering International Conference (BMEiCON)*, pages 1–5, 2016.

[76] S.-H. Park, S. Cho, and J.-R. Lee. Energy-efficient probabilistic routing algorithm for internet of things. *J. Appl. Math.*, 2014, Special Issue:7 pages, 2014.

[77] S. Pasricha, B. K. Donohoo, and C. Ohlsen. A middleware framework for application-aware and user-specific energy optimization in smart mobile devices. *Pervasive Mob. Comput.*, 20(C):47–63, July 2015.

[78] A. W. Paul Ryan, Terence Smith. Total Energy Model for Connected Devices. Technical report, IEA 4E EDNA, June 2019.

[79] I. Popović, S. Janković, and L. Saranovac. Online power-aware scheduling strategy based on workload power profile measurement. In *2017 Zooming Innovation in Consumer Electronics International Conference (ZINC)*, pages 45–46, May 2017.

[80] L. Qing, Q. Zhu, and M. Wang. Design of a distributed energy-efficient clustering algorithm for heterogeneous wireless sensor networks. *Computer communications*, 29(12):2230–2237, 2006.

[81] T. Qiu, Y. Lv, F. Xia, N. Chen, J. Wan, and A. Tolba. ERGID: An efficient routing protocol for emergency response Internet of Things. *Journal of Network and Computer Applications*, 72:104–112, 2016.

[82] K. K. Rachuri, C. Mascolo, M. Musolesi, and P. J. Rentfrow. SociableSense: Exploring the trade-offs of adaptive sampling and computation offloading for social sensing. In *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*, MobiCom '11, pages 73–84, New York, NY, USA, 2011. ACM.

[83] M. Rahman, H.-J. Hong, A. Rahman, P.-H. Tsai, A. Afrin, Y. Uddin, N. Venkatasubramanian, and C.-H. Hsu. Adaptive sensing using internet-of-things with constrained communications. pages 1–6, 12 2017.

[84] T. Ruan, Z. J. Chew, and M. Zhu. Energy-aware approaches for energy harvesting powered wireless sensor nodes. *IEEE Sensors Journal*, 17(7):2165–2173, 2017.

[85] T. Ruan, Z. J. Chew, and M. Zhu. Energy-aware approaches for energy harvesting powered wireless sensor nodes. *IEEE Sensors Journal*, 17(7):2165–2173, 2017.

[86] C. Ruiz et al. Autonomous inventory monitoring through multi-modal sensing (aim3s) for cashier-less stores. In *BuildSys '19*, page 395–396, 2019.

[87] C. Ruiz, J. Falcao, S. Pan, H. Noh, and P. Zhang. Aim3s: Autonomous inventory monitoring through multi-modal sensing for cashier-less convenience stores. In *6th ACM BuildSys*, page 135–144, 2019.

[88] P. Sarwesh, N. S. V. Shet, and K. Chandrasekaran. Energy efficient and reliable network design to improve lifetime of low power iot networks. In *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pages 117–122, 2017.

[89] P. Silsupadol, K. Teja, and V. Lugade. Reliability and validity of a smartphone-based assessment of gait parameters across walking speed and smartphone locations: Body, bag, belt, hand, and pocket. *Gait & Posture*, 58:516–522, 2017.

[90] P. Sinha and A. A. Zoltners. The Multiple-Choice Knapsack Problem. *Operations Research*, 27(3):503–515, 1979.

[91] L. Song, K. K. Chai, Y. Chen, J. Schormans, J. Loo, and A. Vinel. Qos-aware energy-efficient cooperative scheme for cluster-based iot systems. *IEEE Systems Journal*, 11(3):1447–1455, Sept 2017.

[92] E. E. Stone and M. Skubic. Fall detection in homes of older adults using the microsoft kinect. *IEEE J. of Biomedical and Health Informatics*, 19(1):290–301, 2015.

[93] A. Subramaniam, H.-X. Tan, and H.-P. Tan. Ble-enabled medication events monitoring system (mems) for community dwelling seniors. In *2019 International Conference on Information Networking (ICOIN)*, pages 227–232, 2019.

[94] K. Suresh et al. Eeiot: Energy efficient mechanism to leverage the internet of things (iot). In *ICETT*, pages 1–4, 2016.

[95] X. Tan, H. Zhao, G. Han, W. Zhang, and T. Zhu. Qsdn-wise: A new qos-based routing protocol for software-defined wireless sensor networks. *IEEE Access*, 7:61070–61082, 2019.

[96] C. Tang and C. S. Raghavendra. Wireless sensor networks. chapter Compression Techniques for Wireless Sensor Networks, pages 207–231. Kluwer Academic Publishers, 2004.

[97] M. Y. S. Uddin et al. The SCALE2 multi-network architecture for iot-based resilient communities. In *IEEE International Conference on Smart Computing*, pages 1–8, 2016.

[98] W. Wang, P. Mishra, and S. Ranka. Dynamic cache reconfiguration and partitioning for energy optimization in real-time multi-core systems. In *2011 48th ACM/EDAC/IEEE Design Automation Conference (DAC)*, pages 948–953, June 2011.

[99] Y. Wang, J. Lin, M. Annavaram, Q. A. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh. A framework of energy efficient mobile sensing for automatic user state recognition. In *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services*, MobiSys '09, pages 179–192, New York, NY, USA, 2009. ACM.

[100] A. Warrier et al. How much energy saving does topology control offer for wireless sensor networks? - a practical study. *Comput. Commun.*, 30(14-15):2867–2879, 2007.

[101] A. D. Wood, J. A. Stankovic, G. Virone, L. Selavo, Z. He, Q. Cao, T. Doan, Y. Wu, L. Fang, and R. Stoleru. Context-aware wireless sensor networks for assisted living and residential monitoring. *IEEE Network*, 22(4):26–33, July 2008.

[102] L. Yang, R. P. Dick, H. Lekatsas, and S. Chakradhar. Online memory compression for embedded systems. *ACM Trans. Embed. Comput. Syst.*, 9(3):27:1–27:30, Mar. 2010.

[103] M. Yarvis et al. Exploiting heterogeneity in sensor networks. In *Proceedings IEEE 24th Annual Joint Conference of the IEEE ComSoc.*, volume 2, pages 878–890, 2005.

[104] M. Ye, C. Li, G. Chen, and J. Wu. Eecs: an energy efficient clustering scheme in wireless sensor networks. In *PCCC 2005. 24th IEEE International Performance, Computing, and Communications Conference, 2005.*, pages 535–540, April 2005.

[105] D. Zeng et al. Energy minimization in multi-task software-defined sensor networks. *IEEE Transactions on Computers*, 64(11):3128–3139, Nov 2015.

[106] M. Zhao, I. W. H. Ho, and P. H. J. Chong. An energy-efficient region-based rpl routing protocol for low-power and lossy networks. *IEEE Internet of Things Journal*, 3(6):1319–1333, Dec 2016.

[107] Q. Zhu, M. Y. Sarwar Uddin, N. Venkatasubramanian, and C.-H. Hsu. Spatiotemporal scheduling for crowd augmented urban sensing. In *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, pages 1997–2005, 2018.

[108] Q. Zhu, M. Y. S. Uddin, Z. Qin, and N. Venkatasubramanian. Upload planning for mobile data collection in smart community internet-of-things deployments. In *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*, pages 1–8, 2016.

Appendix A

Energy Optimization for IoT: Proof of Complexity

We prove the computational complexity (NP- hardness) of the energy optimization for heterogeneous IoT devices problem, by showing that the minimum multiple-choice knapsack problem, which is known to be NP-complete, can be reduced to it. Knapsack problem has been widely studied in computer science for years. It is one of the problems on Karp's original list of 21 NP-complete problems [51]. There exist several variants of the problem, in this proof we use the minimum multiple-choice version of the problem.

The formulation of the energy optimization for heterogeneous IoT devices problem is:

According to our settings,we have a set of n heterogeneous IoT devices in a certain segment, $i = 1, \cdots, n$. Each device i can be described by a *profile*, which consists of multiple configurations k.

e_{ik} and a_{ik} denote the rate of energy consumption and accuracy level respectively for configuration k of device i $(1 \leq k \leq l_i)$. Each device has a remaining battery capacity, denoted by r_i, at a certain time. Note that for wall-powered devices r_i is not defined or assumed to be ∞.

We prove this constraint optimization problem, formulated as equation (8). with all information available, is an NP-hard by showing that the minimum multiple-choice knapsack problem which is known to be NP-complete, can be reduced to it.

In order to define the minimum multiple-choice knapsack problem formally, consider m mutually disjoint classes$N_1 \cdots N_m$, $i = 1, \cdots, m$ of items to be packed into a knapsack to be at least the capacity C. Each item j in i class has a cost c_{ij} and a size s_{ij}, and the problem is to find a subset of exactly one item from each class such that least profitable set of items such that the total size of the selected items is at least the capacity C. If we introduce the binary variables x_{ij}, which take on value 1 if and only if item j is chosen in class i, the MMCKP is formulated as:

$$\begin{aligned}
\text{minimize} \quad & \sum_{i=1}^{m} \sum_{j \in N_i} x_{ij} \cdot c_{ij} \\
\text{subject to} \quad & \sum_{i=1}^{m} \sum_{j \in N_i} x_{ij} \cdot s_{ij} \geq C \\
& \sum_{j \in N_i} x_{ij} = 1, \forall i
\end{aligned} \tag{A.1}$$

The energy optimization for heterogeneous IoT devices problem belongs to NP as there is a subset of IoT devices with selecting at most one configuration from each of n IoTs ,device i has l_i different configurations, that has the least total cost, and the total combined-accuracy is more than or equal to the activity's accuracy demand τ. Thus, if we have a proposed correct "yes" solution, we can verify this solution in polynomial time $O(n)$ by checking that energy optimization for heterogeneous IoT devices problem has a satisfying subset.

Reduction from minimum multiple choice knapsack problem $<_P$ energy optimization for heterogeneous IoT devices problem. In other words, minimum multiple-choice knapsack problem is polynomial reducible to the energy optimization for heterogeneous IoT devices problem. We consider an instance of minimum multiple-choice knapsack problem, and we will construct an equivalent instance of the energy optimization for heterogeneous IoT devices problem.

- m mutually disjoint classes in minimum multiple-choice knapsack problem $\rightarrow$ n IoT devices each with multiple configurations in energy optimization for heterogeneous IoT devices problem.
- Class N_i has multiple j items $\rightarrow$ Device i has l_i different configurations.
- c_{ij} cost for each item j in class N_i $\rightarrow$ c_{ik} cost for each IoT device i in its configuration k
- s_{ij} size of each item $\rightarrow$ (-ln$(1 - a_{ik})$) accuracy delivered by each IoT device i in its configuration k, note that this is $(-\ln(1 - a_{ik}))$ a positive value as $0 \leq a_{ik} \leq 1$
- C the least capacity limit $\rightarrow$ (-ln$(1 - \tau)$) demanded accuracy

Therefore, the Minimum multiple-choice knapsack problem can be reduced to energy optimization for heterogeneous IoT devices problem, which means energy optimization for heterogeneous IoT devices problem is at least as hard as the Minimum multiple-choice knapsack problem. Therefore, the energy optimization for heterogeneous IoT devices problem is NP-hard.