Emerging Sensing Applications and Protocols for the Internet of Things

Emerging Sensing Applications and Protocols for the Internet of Things

By

Shama Siddiqui, Anwar Ahmed Khan and Indrakshi Dey

Cambridge Scholars Publishing

Emerging Sensing Applications and Protocols for the Internet of Things

By Shama Siddiqui, Anwar Ahmed Khan and Indrakshi Dey

This book first published 2023

Cambridge Scholars Publishing

Lady Stephenson Library, Newcastle upon Tyne, NE6 2PA, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

Copyright © 2023 by Shama Siddiqui, Anwar Ahmed Khan and Indrakshi Dey

All rights for this book reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owner.

ISBN (10): 1-4438-0810-5
ISBN (13): 978-1-4438-0810-1

CONTENTS

PREFACE

A network of objects that have unique addresses has become popular today as the so-called ***Internet of Things*** (IoT). Such objects can be machines, animals, or even human beings. These objects are principally integrated with sensors and actuators that can share information over a unified framework where the information can be processed across different software platforms and protocols, thereby enabling different emerging innovative applications. Different platforms can operate using different data analysis and processing techniques, while the processed data can be used to reach a consensus or decision on a particular phenomenon through a central cloud computing platform. IoT is enabled through cooperation and sharing between a group of so-called 'smart objects,' comprising smart sensors and smart actuators. Smart objects, in turn, refer to devices that have some computational and communication capabilities. When connected, such smart objects create an Internet-like structure, where each object is uniquely identifiable by its virtual address within the network.

The three component technologies that enable IoT are the: a) networks of sensors and actuators or any other communication entities, b) software layers that bridge the computer applications with the actual hardware of the communication entities, enabling real-time input-output communications, and c) cloud computing platforms capable of storing large amounts of data and processing them for activating relevant actions. This book deals with the network of devices, smart things, objects, their underlying protocols, and key applications. The book begins by introducing the fundamentals of IoT concepts, architecture, requirements, and applications in the first chapter. The subsequent six chapters cover sensing applications, including environmental, agricultural, livestock, industrial, medical, vehicular, and threat detection/sensing. Each chapter provides a detailed account of the frameworks, architectures, and protocols used for each application along with their specific challenges. Finally, the book closes by describing the

open research challenges and future opportunities in the field of IoT sensing.

Chapter 1 starts by providing an overview of IoT networks and the work done so far towards their design, development, and deployment. Next, it introduces readers to the generic architecture and fundamental components that form the integral parts of an IoT network. It also provides an overview of the different standards and protocols that have been developed and borrowed from other networks for the proper functioning of an IoT network. Next, it briefly introduces the basic characteristics required for an IoT network to operate efficiently. The chapter concludes by summarizing the major application areas and use cases, and the emerging trends in IoT networks.

Chapter 2 details different environmental sensing applications as a part of the IoT network structure. It provides an overview of the different kinds of sensors used and their corresponding applications, like environment protection, water pollution and air quality monitoring, and smart farming. Next, it introduces different IoT protocols that can be used for environmental sensing and monitoring applications. How present environment sensing functionalities can be optimized is also elucidated. Chapter 2 concludes by enlisting the challenges with present environmental sensing applications.

Chapter 3 introduces and details the applications and protocols related to agriculture and livestock sensing. Agricultural and livestock sensing possess challenges different from traditional IoT sensing applications as they deal with living animals and crops. The chapter also elaborates on designing smart agriculture and livestock management systems. Chapter 3 concludes by summarizing the emerging trends and challenges in agriculture and livestock sensing applications.

Chapter 4 starts with detailing the evolution, enabling technologies, and functional components of industrial IoT and Industry 4.0. It then provides an overview of the various kinds of industrial sensors used for automating industrial operations. The protocols developed so far for industrial sensing

applications are summarized, and how these applications affect performance, production, and safety in the industrial automation environment are elaborated. Finally, chapter 4 sheds light on the different challenges associated with present industrial sensing technologies.

Body sensing networks (BSN), or the Internet of Medical Things (IoMT), is a network of devices (sensors/relays/receivers) that can be deployed on a human body or within a human body to monitor physiological parameters, manage the condition of patients, and predict possible emergencies and activate the relevant response. Chapter 5 details the different applications and protocols enabling the applications for intra-body, on-body, and inter-body sensing. The design requirements, benefits, emerging trends, and challenges related to IoMT are also summarized in this chapter.

A network of vehicles, like cars, trains, planes, and unmanned ariel vehicles (UAVs), and roadside infrastructures like traffic signals, lights, and cellular base stations is referred to as the Internet of Vehicles (IoVs). IoVs present quite different challenges compared to traditional IoT networks as IoVs deal with highly mobile communication entities. Chapter 6 elaborates on the fundamental components and basic architectures of IoVs. Different sensing technologies and routing protocols developed, so far, for IoVs are also detailed. How vehicular sensing applications affect security and privacy involved with IoVs are discussed in chapter 6, which concludes the challenges and emerging trends associated with vehicular sensing applications.

Chapter 7 elaborates on the different applications related to threat detection and sensing, the motivations behind those applications, and how they can be relevant to industrial environments, healthcare facilities, intrusion detection systems, and their operation. The challenges faced by threat detection applications in IoT networks are also summarized in chapter 7.

Finally, chapter 8 presents some open challenges and research directions associated with IoT sensing and applications, such as bandwidth utilization, energy efficiency, delay minimization, reliability maximization, and threat minimization.

CHAPTER 1
Fundamentals of the Internet of Things (IoT) and Sensing

1.1. Introduction

The Internet of Things (IoT) refers to the massive ad-hoc networks connecting physical objects, machines, and humans with the Internet. The ubiquity of the wireless network infrastructure has enabled millions of devices to collect and share data in real-time via the Internet. The autonomous nature of IoT devices and their capability to continue operating for long periods without the need for human intervention has been the major attraction for a diverse range of users [1]. Today, IoT solutions are being deployed for various sensing applications, including, but not limited to, the domains of environment, agriculture and habitat monitoring, industry, medicine, autonomous vehicles, smart city, security, and many more. According to an estimate made by Statista, the number of connected devices will reach 30.9 billion by the end of 2025 [2]; to support the ever-increasing number of connected devices, innovative and efficient technologies and protocols are being developed at a rapid pace.

Due to the advancements brought in micro-electromechanical systems (MEMS) and wireless communication technology, it has become possible to transform any physical object into its "smart" version. According to the application requirements, any object/thing, which could be as small as a pill or as big as an elephant, can be connected to the Internet. Embedding physical objects with sensors and communication hardware enables them to communicate with the Internet, other objects, and humans in real-time.

The major focus of this book is on emerging sensing applications of IoT. The most efficient protocols and architectures/frameworks for each application have been presented, along with the open challenges. This chapter first describes the fundamental concepts of IoT, followed by a discussion on basic design, architecture, requirements, and challenges.

1.2. History and Evolution

The concept of IoT dates to the innovation of the wired telegraph in 1830 when scientists enabled machines to communicate with each other. In June 1900, the first radio voice transmission became a reality, which provided the foundation for wireless telegraphy or communication. In the early 1970s, Theodore Paraskevakos demonstrated the concept of machine-to-machine (M2M) communication by inventing electric meters that communicated with electricity grids.

Although the term "IoT" was officially introduced in 1999, the concept was realized earlier when the Coca-Cola machine was designed to be accessed through the Internet to check if a certain drink was available; the system was developed at Carnegie Melon University and the customers could check the availability and temperature of the drink before making a physical visit to the dispenser. Furthermore, in 1990, John Romkey demonstrated an electric toaster that was controlled via the Internet at the INTEROP conference. In 1999, Kevin Ashton, then executive director of the Auto-ID Center, introduced the concept of the "Internet of Things" while describing the radio frequency identification (RFID) supply chains for Procter & Gamble. This was when industrial professionals first began to explore the possibilities of using connected things for diverse application scenarios.

In 2000, the idea of a smart (Wi-Fi compatible) refrigerator was put forward by LG, with which the era of smart household objects began. The smart refrigerator allowed customers to order food online and make video calls. Starting in 2003, companies began using the term "IoT" for the concept previously known as M2M. In 2005, a small rabbit-shaped robot was introduced that provided details about the present stock exchange rates, the latest news, and weather forecasts. Gradually, the number and type of devices connected to the Internet kept growing, and today we see multiple devices such as security cameras, point of sales (PoS) terminals, GPS trackers, and fitness monitors connected to the Internet using common Wi-Fi or cellular network connections.

The concept of IoT began to gain wide popularity in 2013 with the emergence of cutting-edge wireless communication and embedded systems technologies. Since then, IoT has been supported by several technologies,

including wireless sensor networks (WSN), control systems, the Global Positioning System (GPS), and so on. Standard as well as dedicated protocols, and hardware and networking solutions have been proposed to meet the ever-growing need of the IoT market worldwide.

1.3. Architecture and Components

Fundamentally, the IoT network comprises smart devices or things that are programmed to collect data from the environment and transmit it to a central server or dashboard via a gateway. The IoT architecture defines physical components and their organization, network configuration, data format, and operational procedures, which must have design compatibility to achieve the overall objective of the IoT solution. Familiarity with IoT components and architecture is essential for developing an understanding of IoT application scenarios, protocols, and challenges.

1.3.1. *Generic Architecture*

A generic IoT architecture is illustrated in figure 1.1. At the lowest layer, IoT comprises various distributed sensor nodes that are configured to collect data periodically or sporadically from the environment. These sensors may collect data about any environmental or physical/physiological parameter. The second layer comprises cluster head and gateway nodes. Various sensor nodes transmit data to the cluster head for local processing, which is subsequently forwarded to the gateway. Either whole data or aggregated values may be transmitted to the cluster head/gateway. For certain IoT applications, handheld devices such as smartphones may serve as a gateway. Finally, the data is taken to the cloud where it is stored in the databases. From this point, the data can be accessed by remote physicians or any other stakeholder based on the application requirements. Each component of IoT architecture is briefly explained in the next paragraphs.

1.3.1.1. IoT Device/Sensor or Actuator Node

Sensors exist at the lowest layer of the IoT hierarchy. These sensors may be embedded with the physical objects to transform them into smart objects or may facilitate sensing activity independently. The type and use of the sensor

largely differ based on the application requirements. Moreover, actuators are also often used by IoT solutions in integration with the sensors. For example, for the healthcare IoT, an implanted body sensor network (BSN) may use both the sensor and actuator nodes, where the sensors could be

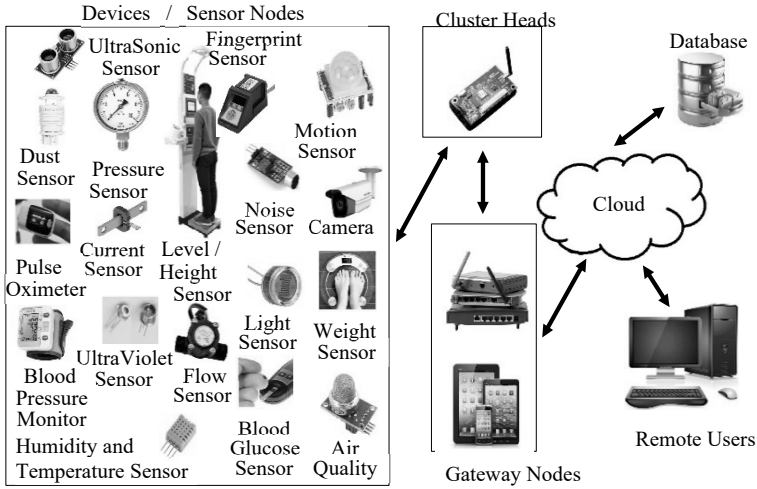

Figure 1.1 General IoT architecture depicting connection of sensor nodes with remote stations

configured to sense and transmit health data regularly, and the actuator may be programmed to inject the medicine inside the human body upon receiving an alert signal from the remote location [3].

1.3.1.2. IoT Cluster Head

Most IoT devices are resource-constrained, and it is not practical to do heavy computations at each sensor node. Instead, data from multiple sensor nodes are collected at a single cluster head that possesses higher power, storage, and computation capabilities. The cluster head often aggregates data received from multiple sensor nodes and forwards the consolidated data to the next tier of IoT architecture to conserve energy and reduce possible network congestion. Moreover, the cluster head also provides other facilities to the nodes, such as synchronization, scheduling of transmission slots, informing about path breaks, handling network scalability, etc.

1.3.1.3. IoT Gateway/Edge Node

Edge nodes perform computing in a distributed manner so that the burden of the central node/remote server decreases. Data from cluster heads are transmitted to the edge node, which could be a wearable or hand-held device. The sensor and gateway nodes can function in a low-bandwidth mode. Edge nodes are closer to the sensor nodes, and may provide a quicker response to the data shared by sensors compared to receiving instructions from the remote devices. Hence, the edge nodes often transmit the selected/aggregated data to the remote server instead of forwarding all packets received from sensor nodes. In this way, the pre-processing and filtering of data improves the efficiency of the IoT solution by reducing the costs associated with the transmission, processing, and storage.

Various techniques are utilized to ensure the smooth operation of edge nodes. Firstly, the total operational overhead cost for each edge node is estimated by considering the requirements of power consumption and regular upgrades; this helps to identify the lifetime value of each node. Secondly, the security of edge nodes is ensured by deploying encryption and firewall technologies. Thirdly, bandwidth maximization strategies such as data aggregation are used to ensure resource optimization. Furthermore, techniques for anomaly detection have also been proposed to ensure that the resources of the edge node are not wasted due to processing fake data [4]. Today, the edge node has been regarded as an excellent means for providing unique services such as Ambient Assisted Living (AAL) due to timely and reliable data transmission. AAL offers the patients a chance of continuous remote monitoring while the remote physicians also remain aware of the patient's health history, diagnosis, and therapeutic requirements.

1.3.1.4. Remote Server

The gateway/edge nodes transmit data to the remote server. This is the central location that keeps the data stored and accessible for the stakeholders for each IoT application. The data are often stored in cloud servers where appropriate security solutions are also implemented to protect all the IoT stakeholders.

1.3.2. *Layered Architecture*

Although IoT architecture is highly flexible, it is commonly defined as a three-layered architecture. It comprises perception, network and application layers, as shown in figure 1.2.

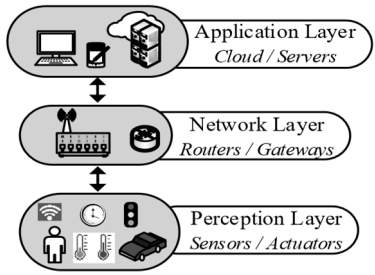

Figure 1.2: IoT layer architecture

1.3.2.1. Perception Layer

This refers to the physical layer of IoT architecture and comprises sensors and other devices that interact directly with the environment. As described previously, the sensors collect and share data according to the requirements and configurations. In addition to sensors, edge devices and actuators also belong to this layer.

1.3.2.2. Network Layer

The network layer is responsible for processing and transmitting the data collected by the perception layer. This layer provides connectivity between the perception layer devices and other smart devices, such as network servers and handheld devices.

1.3.2.3. Application Layer

This layer offers a user interface for the IoT solution and provides application-specific services/data to the users. The users may choose to perform various actions by interacting with the application layer. For example, for a health monitoring application, the users may check a log of

their blood pressure, and for a home automation system, the users may turn their coffee machine on remotely.

1.4. IoT Standards and Protocols

Several standards and protocols have been established to support the data collection and transfer from IoT hardware to the designated locations in an organized manner. The IoT protocols enable interaction between various components of IoT architecture and ultimately facilitate the timely transmission of meaningful data while maintaining all the network resources. In short, a protocol can simply be regarded as a language for all IoT devices. This section reviews some of the major standards and protocols used by the IoT stack.

1.4.1. *Constrained Application Protocol (CoAP)*

Although the worldwide Internet infrastructure is accessible and available for all IoT devices, for certain use cases of IoT, using the Internet may not be possible due to resource limitations. Considering particularly the power constraints, the Constrained Application Protocol (CoAP) was designed by the IETF Constrained RESTful Environments working group in 2013. CoAP mainly targets translating the HTTP model in a way that it becomes more adaptable for restrictive network and device environments.

CoAP has been built on User Datagram Protocol (UDP) for addressing the needs of HTTP-based IoT systems and establishing end-to-end secure connections. UDP facilitates the data transmission to multiple hosts by allowing both multi-casting and broadcasting modes; UDP serves as the best choice for wireless networks because of its low bandwidth requirement and communication speed, and hence it has been used by CoAP for deployment in the M2M networks. RESTful architecture is another feature shared by HTTP and CoAP; this architecture supports the request/response mechanism between the hosts. Finally, CoAP even deploys the basic get, post, put, and delete mechanisms of HTTP. Clearly, these features allow CoAP to work with machines seamlessly while maintaining ambiguity at the lowest level.

The quality of service (QoS) feature has also been integrated with CoAP as it inherently controls the transmitted messages by marking them

as “confirmable” non-confirmable”; this categorization of messages helps the receivers to know whether they need to send back an acknowledgement frame. CoAP also supports the mechanisms of resource discovery and content negotiation. In addition to transmitting the data generated by IoT devices, CoAP also participates in securely transmitting the transport layer messages by deploying the Datagram Transport Layer Security (DTLS). Therefore, CoAP has been designed to fully serve as a lightweight IoT protocol that functions on energy- and resource-constrained IoT devices. This protocol is particularly useful for developing IoT solutions based on web services.

1.4.2. *Message Queuing Telemetry Transport (MQTT)*

Message Queuing Telemetry Transport (MQTT) has been regarded as one of the major lightweight wireless protocols used recently for industrial IoT (IIoT) applications. This protocol is based on the publication/subscription (pub/sub) model. The architecture of MQTT has also been designed to be lightweight so it can run on battery-operated devices for long periods. Since MQTT was designed on top of TCP, unlike CoAP, this protocol is often advocated for unreliable application areas.

MQTT uses the model of publisher, subscriber, and broker. In this model, the publisher is responsible for collecting data and transmitting it to all the subscribers using the broker as a mediating layer. The broker is mainly responsible for ensuring the security of the network, and it authenticates the publishers and subscribers by cross-checking their credentials. MQTT offers QoS using three different modes, which can be chosen based on the criticality of the application.

- ***QoS0 (At most once)***: In this mode, the publication is sent but confirmation is not received. It is the least reliable but fastest mode for MQTT operation.
- ***QoS1 (At least once)***: This mode ensures that the message is delivered at least once, however, duplicate messages may also be received which may cause energy depletion.
- ***QoS2 (Exactly once)***: In this mode, the message is delivered only once, and the duplicates are suppressed. As a result, this mode becomes the most reliable, but it is also the least bandwidth efficient.

MQTT has been designed to work well with various IoT devices, including but not limited to vehicles, detectors, electric meters, sanitary and conventional industrial equipment, and sensing units. MQTT can be configured to work with the minimum bandwidth, least energy, required level of reliability, and minimum memory and processing resources.

On the negative side, MQTT may pose challenges for devices with very low computational resources due to using TCP. To resolve this inherent design issue, a variant of MQTT that works with UDP has also been introduced and is referred to as MQTT-SN. MQTT-SN supports topic name indexing and avoids long topic names, which was another issue with MQTT. However, when compared with MQTT, MQTT-SN does not offer well-defined data representation, and the model used for the device management structure is not as efficient. Mostly, it is considered that MQTT-SN provides platform- or vendor-specific data and device management functionalities.

1.4.3. *IEEE 802.11 Wi-Fi*

A Wi-Fi IEEE 802.11 standard network is created when devices such as computers, telephones, and routers send wireless signals to each other. Today, Wi-Fi has commonly been used for connecting to the Internet wirelessly. In indoor environments like offices or homes, a router is often used to transfer the Internet connection from the public to a private network. Thus, the router provides an Internet connection to all the nearby devices by advertising the availability. Moreover, Wi-Fi hotspots are also created to share their wireless or wired Internet connection.

Wi-Fi technology employs radio frequencies such as 2.4 GHz or 5 GHz for broadcasting wireless signals. Each of these frequency ranges has a number of channels that may be used by different devices; the distribution of ranges in channels helps to distribute the load so individual connections may not be interrupted due to the increased network load. Hence, the spectrum is shared in such a way that wireless networks could avoid traffic overflow to a large extent.

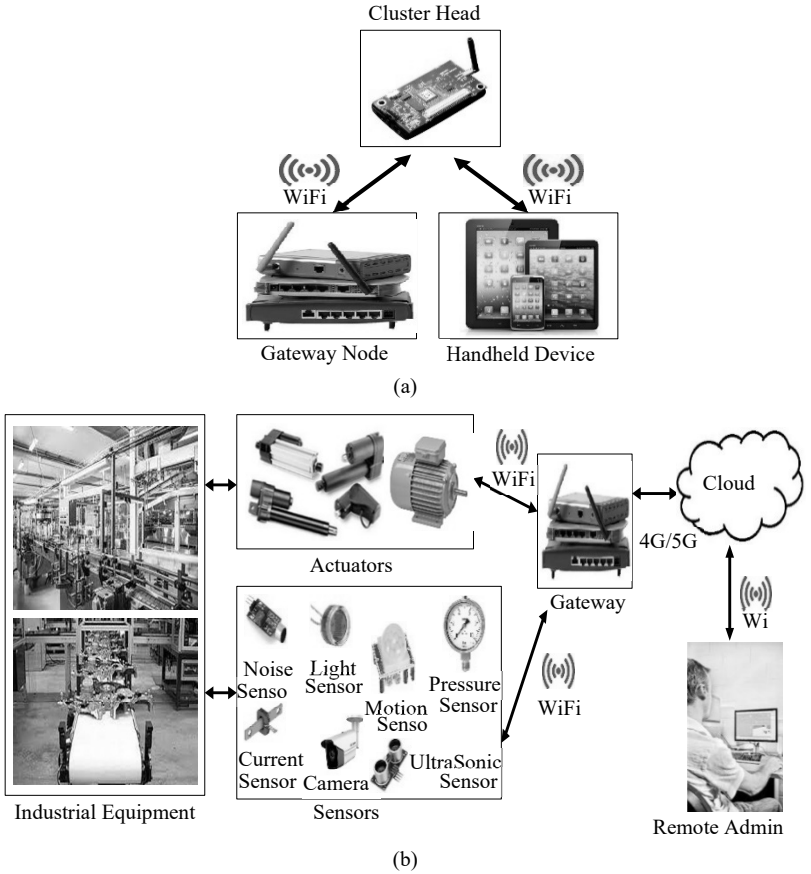

Figure 1.3: Use of Wi-Fi for connecting IoT devices to the remote stations via a gateway.
(a) Wi-Fi connecting IoT cluster head with handheld device or router where both act as a gateway
(b) Conventional scenario of industrial IoT where remote admin controls the industrial equipment

Typically, Wi-Fi works smoothly over a distance of 100 metres. However, in most scenarios, the operating distance for Wi-Fi is restricted to 10–35 metres to guarantee efficiency. Similarly, factors such as operating frequency, antenna strength, and environmental interference also largely affect the transmission quality and effective coverage of Wi-Fi. This is the

major reason why transmission speed may increase as the connected device reaches the source.

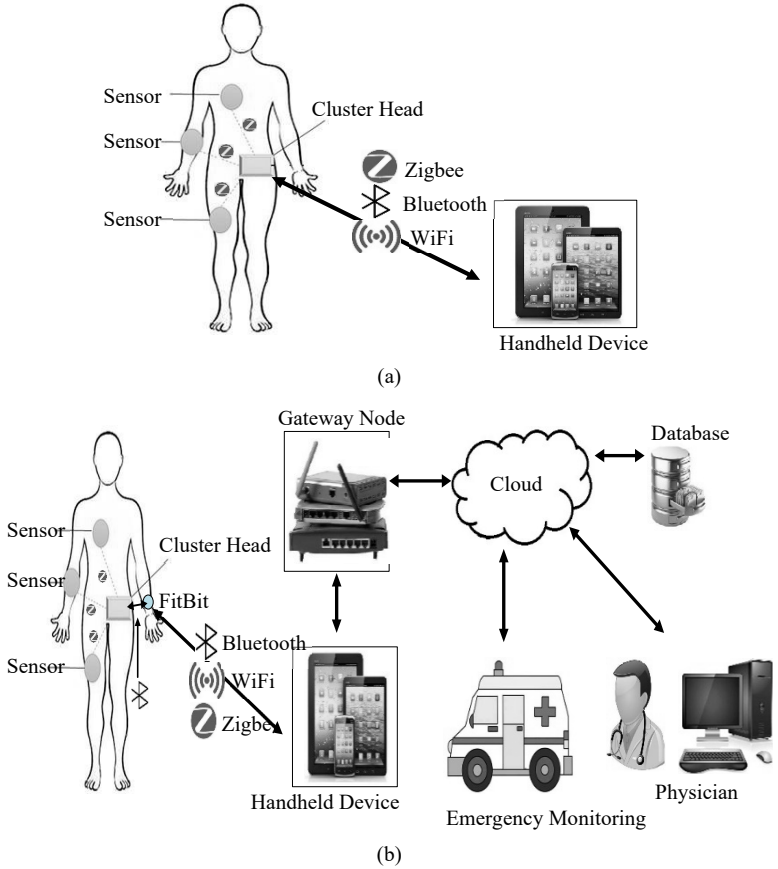

Figure 1.4: Short range communication standards for medical IoT applications. (a) Cluster head communicating directly with handheld device using Wi-Fi, Zigbee, or Bluetooth. (b) Cluster head communicating with Fitbit using Bluetooth, which subsequently communicates with a handheld device

Wi-Fi has been used for IoT solutions for the transmission of data from the cluster head node to the hand-held device, and from the hand-held device to the router. Hence, Wi-Fi is deployed both for transmitting data from the IoT device to the central remote location and for sending signals

back from remote locations to the IoT device, as shown in 1.3(a). For example, a remote manager may use Wi-Fi coupled with other wireless networking technologies to control an industrial actuator for controlling certain equipment. An example scenario employing Wi-Fi is depicted in figure 1.3(b).

1.4.4. *Zigbee*

Zigbee was also designed for networks that are low power, require low throughput (up to 250 kbps), and have a low connectivity range (up to 100 metres). Most common applications using Zigbee include wireless sensor networks, home automation systems, alarm and alert systems, remote monitoring systems, wireless personal area networks (WPANs), and wireless body area networks (WBANs).

The initial specification of Zigbee was recognized as an IEEE standard in 2003. This standard was developed mainly for short-range radio networks requiring self-configurations, such as telemetry and remote monitoring systems. Zigbee serves as one of the simplest, error-resistant, and secure protocols. It is often used for IoT devices with basic requirements, such as sensing units or microcontrollers, and those that need to be connected in a grid topology. The protocol is particularly easy to deploy and maintain because it was designed to self-configure and self-maintain. Scalability is another major feature offered by Zigbee, which also makes it one of the best choices for IoT vendors, who often design IoT devices using this open standard. Typical usage of Zigbee for WBAN is illustrated in figure 1.4(a), where data from implanted and on-body sensors is communicated to the cluster head using Zigbee, and then data is further taken to a hand-held device using Zigbee, Wi-Fi, or Bluetooth (discussed next).

1.4.5. *IEEE 802.15.1 Bluetooth*

Various electronic devices share their data using Bluetooth technology over short communication distances. Three classes of ERP 1-3 transmission power are offered by Bluetooth standard to serve the open-space transmission range of 100, 10, and 1 metre(s), respectively. The second class has been used the most as it allows sharing data between devices that are at a reasonable distance, and even located in different rooms or on other

floors. Bluetooth uses the 2.4 GHz ISM frequency band, and the device that allows data transmission over this standard is referred to as a Bluetooth adapter.

Data is transmitted using Bluetooth in the form of packets. When the oldest standard of Bluetooth 1.0 is used, packets are transmitted over one of 79 channels. This standard offers a transfer speed of 721 kbit/s with a bandwidth of 1 MHz. On the other hand, if the latest standard 4.0 of Bluetooth is used, a bandwidth of 2 MHz is offered with the provision of 40 channels; as a result, the maximum data transfer speed obtained is up to 3 Mb/s. To ensure a smooth transition, the latest Bluetooth standards have been designed to be compatible with the older versions.

Bluetooth is often used in IoT solutions. For many scenarios, the communication link between the cluster head of the sensor network and the hand-held device or local node has been established using Bluetooth as these are often located in close proximity. Most IoT sensing nodes are by design integrated with the Bluetooth module, which helps users collect data from their devices over their smartphones or any other specialized device. An example scenario where fitness data is collected from the Fitbit/smartwatch over a person's smartphone using Bluetooth is depicted in figure 1.4 (b). This data is subsequently sent to the remote monitoring units via a gateway.

1.4.6. *Long Range WAN (LoRaWAN)*

LoRaWAN belongs to the category of non-cellular Low Power Wide Area (LPWA) protocols. LoRaWAN builds on the radio modulation technology LoRa. The major feature that makes LoRaWAN particularly useful for IoT applications is its long range. Using this protocol, any IoT device or gateway becomes accessible using the single-hop transmission mode, which enables efficient and reliable remote monitoring and control. For example, a user may control their domestic appliances from a distance via LoRaWAN. At the time of the development of LoRaWAN, other non-cellular LPWA solutions, such as IoT for mobile (IoT-M), were either not available or were very expensive. LoRaWAN has a long history of providing a communication infrastructure to IoT solutions compared to other non-cellular solutions. This protocol operates at lower frequency

ranges (unlicensed spectrum) compared to cellular networks. LoRaWAN serves as the best solution for a number of IoT scenarios, such as those that are uplink-oriented, non-critical, delay-tolerant, generate low traffic, are battery-powered, and use low-cost sensors. Lately, LoRaWAN has been used for smart city initiatives across Europe for applications, including smart metering, waste management, farming, logistics, mining, manufacturing, and security.

1.4.7. *SigFox*

SigFox emerged as one of the major competitors for LoRaWAN as both protocols focus on providing communication methods to IoT deployments. SigFox is primarily known as a narrowband or ultra-narrowband communication technology that uses Binary Phase Shift Keying (BPSK) for data encoding. Since the receivers using SigFox only listen for a short while in the spectrum, the protocol is able to deal with interference to a large degree. For efficient SigFox communication to happen, the endpoint radio may be inexpensive, but the base station is expected to be sophisticated. The performance of SigFox is better for the uplink communication (from the endpoint to the base station). Although SigFox, by design, is capable of providing two-way communication, its capacity is constrained for the communication process initiated from the base station, compared to LoRaWAN. Moreover, when compared with LoRaWAN, the spectrum used by SigFox is lower, which may result in better interference management; however, due to better coding gains in LoRaWAN, both protocols practically provide similar levels of link budget.

1.4.8. *Narrow-Band IoT (NB-IoT)*

NB-IoT is another low-cost standard developed for LPWA applications to support a wide array of IoT solutions and devices. It has been designed to co-exist with 3G/4G/5G mobile networks, which implies that data from IoT devices is transmitted over existing mobile network infrastructures. As a result, in addition to increased accessibility, the protocol benefits from the privacy and security features offered by mobile networks, such as user and mobile equipment identification, authenticity, data integrity, and

confidentiality. NB-IoT offers an ideal solution for applications that require low bandwidth, a small amount of data transmission, and a long battery life.

Over the years, NB-IoT has improved on many features, including system capacity, power consumption, and spectrum efficiency. The major attraction of NB-IoT is its long battery life, which is expected to last as long as 10 years for various IoT uses that require far-off and deep coverage. NB-IoT offers excellent coverage in buildings and underground environments, due to which it has been recommended for applications such as remote health monitoring. Since NB-IoT was designed to focus on the requirement of extensive coverage in both the rural and urban environments while ensuring deep indoor connectivity, novel physical layer channels and signals were developed. This is because the earlier physical layer solutions could not be used to realize the deployment of NB-IoT, which maintained an ultra-low level of complexity and extremely low power consumption.

1.4.9. *Long-Term Evolution for Machines (LTE-M)*

LTE-M is a low-power communication technology which also belongs to the LPWA category. Although 4G offers a high degree of bandwidth, reliability, and speed, there are many IoT solutions that do not need extensive resources. Therefore, LTE-M, instead of a cellular network, is preferred for common IoT solutions such as smart metres, alarm systems, asset trackers, industrial sensors, wearable sensors, and smart city controllers. Using LTE-M, IoT devices can connect directly to the mobile (4G) network without needing a gateway. This technology also offers a battery life of nearly 10 years. Just like NB-IoT, LTE-M benefits from the security, authenticity, and integration features of mobile networks. However, the component cost for LTE-M is lower because the chips have a narrower bandwidth and offer half-duplex communication. The technology uses specific methods for saving energy consumption; when an IoT node is not required, it is sent into a deep sleep state known as “Power Savings Mode” (PSM). There is also another mode for the purpose referred to as extended discontinuous reception (eDRX), where the nodes wake up periodically. Variable data rates are also offered to cater to the needs of diverse emerging IoT solutions.

The communication service cost is also quite low for LTE-M since the maximum required data rate is only up to 100 kbps. LTE-M serves as the best choice for various scenarios; for example, any business (such as pharmaceuticals) requiring continuous monitoring of their assets or processes may deploy LTE-M to save the hurdle of installing gateways. Hybrid communication technologies are also being used for asset tracking; in some solutions, Bluetooth is used to establish short-range connections and LTE-M for providing coupling with the backhaul links.

1.4.10. *Extensible Messaging and Presence Protocol (XMPP)*

The Jabber open-source community developed XMPP for real-time messaging in 1999. This protocol is based on XML and is often used for IoT communication middleware. The major attraction of XMPP is that it allows the real-time transmission of structured but extensible data between hosts. XMPP-IoT was developed as a lightweight alternative for IoT solutions that mostly suffer from resource constraints. Since XMPP is an open standard, it has been made increasingly scalable, due to which it seems promising for the massive futuristic deployments of IoT solutions.

There are also several drawbacks to using XMPP in the IoT environment. First, it does not offer any QoS features, and second, it does not offer end-to-end encryption, which has become a crucial need for emerging IoT solutions today (such as in healthcare and industry). Due to these constraints, it has been predicted that XMPP would mostly be confined within the industrial boundaries and could not be used for end-to-end data transmissions.

1.4.11. *Data-Distribution Service (DDS)*

DDS protocol has also been developed using the publish-subscribe model. The Object Management Group designed DDS for real-time M2M communication. The major attraction of DDS is that it provides efficient data transmission between the nodes regardless of the software and hardware platforms. Also, DDS offers high reliability, scalability, and interoperability. In addition, the protocol also offers a high degree of QoS by supporting a multicasting mechanism and brokerless architecture.