Virtual Reality & Intelligent Hardware

• Article •

December 2022 Vol. 4 No. 6: 521–534
10.1016/j.vrih.2022.07.005

Digital twins and multi-access edge computing for IIoT

Andreas P. PLAGERAS, Konstantinos E. PSANNIS*

Department of Applied Informatics, University of Macedonia, 54636, *Greece*

Received 10 April 2022; **Revised** 30 June 2022; **Accepted** 9 August 2022

Abstract: Background All recent technological findings can be collectively used to strengthen the industrial Internet of things (IIoT) sector. The novel technology of multi-access edge computing or mobile edge computing (MEC) and digital twins have advanced rapidly in the industry. MEC is the middle layer between mobile devices and the cloud, and it provides scalability, reliability, security, efficient control, and storage of resources. Digital twins form a communication model that enhances the entire system by improving latency, overhead, and energy consumption. **Methods** The main focus in this study is the biggest challenges that researchers in the field of IIoT have to overcome to obtain a more efficient communication environment in terms of technology integration, efficient energy and data delivery, storage spaces, security, and real-time control and analysis. Thus, a distributed system is established in a local network, in which several functions operate. In addition, an MEC-based framework is proposed to reduce traffic and latency by merging the processing of data generated by IIoT devices at the edge of the network. The critical parts of the proposed IIoT system are evaluated by using emulation software. **Results** The results show that data delivery and offloading are performed more efficiently, energy consumption and processing are improved, and security, complexity, control, and reliability are enhanced. **Conclusions** The proposed framework and application provide authentication and integrity to end users and IoT devices.

Keywords: Digital twins; Energy efficiency; IIoT; Load balance; MCC; MEC; Protocols

Citation: Andreas P. PLAGERAS, Konstantinos E. PSANNIS. Digital twins and multi-access edge computing for IIoT. Virtual Reality & Intelligent Hardware, 2022, 4(6): 521–534

1 Introduction

The industrial Internet of things (IIoT) is a smart network of machines and devices that interact with one another to improve the performance of industrial processes[1]. It requires efficient device-to-device connectivity and communication, increased time-saving, efficient optimization, and secured environment. In the IIoT, different standards, protocols, and technologies deal with different devices and systems. The connections are wired and wireless[2,3].

A challenge that arises in IIoT systems is data collection and transmission, which must be performed with limited energy. Moreover, integration and interoperability issues must be resolved[1]. Another relevant issue is the connectivity between devices[2]. Different IoT devices use different protocols to communicate because

***Corresponding author**, kpsannis@uom.edu.gr

each device produces different types of data. The challenges and issues that are considered to be crucial for meeting sustainability needs include availability, scalability, reliability, response time, power consumption, security, and cost.

Researchers have claimed that since the number of IoT devices reached 50 billion and the amount of data generated by these devices reached 500 zettabytes, the development of novel networking, storage, and energy-constrained solutions has been increasing. The following are some of the technologies that can be combined to solve the issues that arise in the field of the IIoT.

A core technology that is closely associated with the fourth industrial revolution is edge computing (EC), in which data processing is performed at the edges of a network. Some of the benefits of this technology include low latency, low overhead, and the concurrence of resources. Harikrishna et al. reported that EC can be employed to performs many tasks, such as computing, storage and caching, processing, and distribution of requests, to receive responses with results that have been obtained by the "cloud" [4].

Cloud computing (CC), or simply "cloud", is another technology that plays a vital role in data storage, analysis, and security. In recent years, the utilization of mobile cloud computing (MCC) has increased because everybody uses a smart mobile device. However, the need for more computing power and storage is increasing every day, as mobile devices are energy-constrained and cannot handle a large amount of data produced every second.

To address this, the novel technology of multi-access edge computing or mobile edge computing (MEC) was developed. MEC is the middle layer between mobile devices and the cloud and provides scalability, reliability, MCC, security, and efficient control and storage of resources. It also decreases latency, increases efficiency and data rate, and is easy to configure, enabling the resources of the network to be divided into smaller pieces to make efficient matches and provide the right services. This process is called network slicing or logically isolated network partition[5]. Network slicing provides a dynamic infrastructure to run on different logical networks, which are termed slices, and each slice must handle a specific service.

Another region-based technology is machine learning (ML), which facilitates decision-making through the utilization of big data (BD)[6]. Although it is a beneficial technology, it has several drawbacks in the IIoT era. The research community has focused on several aspects that require improvement, such as optimization and control access problems[7].

ML has two paths that enable efficient BD generation, collection, and analytics. Supervised prediction algorithms are used for a set of data (SetD) to obtain a result (analytics) that is based on this set (SetD = $\{(xz, yz)\ z = 1, \ldots, Z\}$, where Z is the data specified in the code and y is the data (a, b) that are considered for each of the data in x). By contrast, unsupervised algorithms use one objective, x, for the specification of the data (SetD = $\{xz, z = 1, \ldots, Z\}$).

In addition to the two ML techniques, other techniques are available, such as query learning (QL), re-inforcement learning (RL), deep RL, deep learning (DL), and federating learning[8].

These technologies have been utilized for the introduction of digital twins. This concept has been widely examined, and digital twins have formed a communication model that consists of an application layer, a middleware layer, a networking layer, and a physical/object layer. The application layer involves a mobile application and application programming interfaces (APIs) integrated into the application. The middleware layer consists of the cloud server (CS) and mechanisms for the processing and storing of data. The networking layer is enhanced with a set of communication protocols and standards that are integral parts of the model, an interface for the interaction, and an adoption mechanism for the objects (e.g., Wi-Fi, BLE 5, and 5G/6G). The last layer, where objects reside, is equipped with embedded objects and connections between them[9–16].

The purpose of this study is to combine different technologies, protocols, algorithms, and tasks to provide a digital twin as an efficient real-time communication and analytics model characterized by low latency, low overhead, and low energy consumption.

The contributions of this study are as follows. The most common IoT protocols are comparatively analyzed. An overview of the IoT protocols is presented, and significant results are obtained, such as the efficiency of each protocol. A novel framework involving an efficient and interoperable solution for communication over different protocols at the edge of an IIoT network is proposed. The PHP framework established is based on a model–view–controller (MVC) architecture that provides interoperability, real-time and efficient communication, authentication, encryption, memory allocation/deallocation, and energy-saving. Subsequently, a novel task offloading algorithm is proposed and implemented to reduce the runtime of tasks and the energy consumption of IoT devices. The proposed hybrid algorithm reduces overhead, improves traffic congestion, and reduces the time required to respond. Moreover, an IIoT network is established and tested in a simulation environment in conjunction with the application of the novel framework to evaluate the efficacy of the proposition. In addition, communication, energy consumption, and packet loss are examined by using an emulator that runs on a Cooja–Contiki operating system (OS).

The remainder of this paper is organized as follows. In Section 2, related scientific work is discussed. In Section 3, possible algorithmic solutions and a distributed system communication model are presented. In Section 4, a comparative study of IIoT protocols and a solution for the interoperability problems caused by the heterogeneity of the devices in this sector are presented. In Section 5, the proposed IIoT network and framework architectures are discussed. In Section 6, the tests that were performed, evaluation metrics used, and experimental results obtained are discussed. Finally, in Section 7, the conclusions and future research directions are provided.

2 Related work

The main focus in this this study is the biggest challenges that researchers in the field of IIoT have to overcome to achieve a smarter communication environment in terms of technology integration, energy and data delivery efficiency, storage spaces, security, and real-time control and analysis (or digital twins)[9–16].

Aidan Fuller et al.[9] and VanDerHorn et al.[10] provided several definitions of the term “digital twins”. In addition, Aidan Fuller et al. comprehensively analyzed the challenges and the most recent research on digital twins in three big sectors: healthcare, industry, and smart cities[9]. They also presented the technologies used to obtain digital twins, namely, IoT and IIoT, REST and SOAP, CC, ML (supervised learning, unsupervised learning, and DL), databases (MongoDB, Redis, and MySQLi), and data analytics and visualization technologies (analytics, statistics, and artificial intelligence).

He et al. studied sustainable manufacturing and digital twin technologies to improve intelligent manufacturing in terms of cost, quality, productivity, and flexibility[11]. He et al. highlighted the complex structures and severe operating conditions that may increase accident frequencies and maintenance challenges[12]. To secure, automate, and monitor processes in applications and optimize control, they proposed digital twin system applications that have automatic processes.

Qi et al. studied all recent technologies and tools that could enhance the digital twin technology and its applications[13]. They comprehensively analyzed the tools and technologies that could be used for the implementation of digital twins. The results were represented in 5D models for ease of understanding. Azad M. Madni et al. studied the integration of the digital twin technology with IoT, ML, simulation technology, and model-based system engineering[14].

Chi-Hung Hsiao et al. attempted to solve the integration issues in the IIoT by proposing an open-source framework[1]. This web application framework is a communication protocol platform that provides an opportunity for developers to make the right protocol choice, test, enhance security, analyze storage spaces, and solve integration issues. The protocols that have been chosen are the Open Platform Communication Unified Architecture (OPC UA), Message Queuing Telemetry Protocol, Advanced Message Queuing Protocol, representational state transfer, and MTConnect. The beneficial protocol that Chi-Hung Hsiao et al. proposed for use, depending on the data that must be transferred, was the OPC UA.

Sotirios K. Goudos et al. presented an overview of IoT technologies in every layer of an IoT communication model[17]. They also classified layers that constitute an IoT communication model, protocols that can be used in each layer efficiently, and technologies that can support each layer. Moreover, they compared and discussed IoT protocols and technologies. Scientists have concluded that all IoT protocols have a constrained or compressed feasibility, such as the Constrained Application Protocol (CoAP), which is one of the protocols that will be widely used in the future in conjunction with the Hypertext Transfer Protocol (HTTP) that supplies the application layer. 5G networks and the semantic web have also been analyzed. The semantic web provides a solution to the interoperability problem. The layers for the representation of data in such an environment, which are highlighted in this study, consist of the Extensible Markup Language (XML), Resource Description Framework (RDF), Web Ontology Language, and “logic”.

Researchers have presented a framework that utilizes different metrics to measure the quality of service (QoS) and achieve a sustainable IoT scenario. Furthermore, according to the current trends and research, another technology that has been developed jointly with the IoT is CC, which offers excellent opportunities in terms of service management and sustainability.

Researchers have reported the typical QoS requirements of different devices in a smart industry[7]. In addition to the non-critical data produced by sensors and tracking devices that require low latency and low data rates (Kbps), many critical data are produced by devices that require lower latency and higher data rates (Mbps), and these data are given priority over non-critical data.

Syed et al. proposed an EC network architecture that provides few network overheads[5]. They discussed network slicing using different layers and standards. They also highlighted that network slicing is considerably challenging because it supports the QoS requirements for 5G. They showed that the best solution to obtain the most efficient data from all resources is dynamic radio access network slicing with shared resources.

Islam et al. provided an overview of the existing models (centralized, decentralized, and distributed) for offloading network tasks in single and multiple edge servers (ESs)[18]. They also provided metrics for full-stack evaluation. In addition, they performed a comprehensive comparative analysis for the issues addressed, methods applied, algorithms used, various performance metrics, and system models used in each case.

Borsatti et al. proposed an MEC-based architecture in which edge, fog, and CC provide the benefits of computing, storing, and networking closer to the user[19]. To achieve these benefits, the data generated by the IIoT devices were processed at the edge of the network (by local ESs). Consequently, traffic and latency were decreased, and security, control, and reliability were enhanced. They presented a detailed reference scheme for IIoT as a Service, features and components of the framework, and several platforms. Some of the platforms used in that study were OpenStack in the cloud, Kubernetes for container orchestration using Docker as an engine for containers, CoreDNS, Calico for containerized networking, and MetalLB for load equilibrium. The communication protocol used was MQTT.

3 Distributed system communication model

In the first layer (physical) where data are generated, IIoT devices are initialized and connected to the infrastructure. This is the first layer in which data can be managed before, after, and during their generation. Various techniques and algorithms have been proposed, such as waveform techniques, orthogonal frequency division multiplexing, filtered multi-tone mode of filter bank multicarrier, universal filtered multicarrier,

Table 1 Comparative analysis of load balancing algorithms

	RR	WRR	LC	WLC	Random
Servers with identical specifications	√	–		–	√
Supporting critical apps	–	√	–	√	–
Overloaded server√		√/ – If connected for longer period than expected	–	–	√/ – If nodes have different specs

generalized frequency division multiplexing, and load balancing algorithms. As indicated in Table 1, the most common load balancing algorithms include the round robin (RR), weighted round robin (WRR), least connections (LC), weighted least connections (WLC), and random.

Many researchers have been studying task offloading solutions[4,5,18,20−23] for the load balancing and distribution of tasks between ESs. Akhiruh et al. claimed that because of the distance between the local system and cloud, the propagation delay (distance divided by the propagation speed) and failure of synchronization issues due to the increased packet delay deviation (jitter) have been increased between the local devices and CS[18]. The entire task offloading on the CS can cause further expansion of the task completion time[18]. Therefore, a solution to this problem is to offload the task on ESs. This will reduce overhead and improve traffic congestion and the time required to respond.

The purpose of offloading tasks to MEC servers is to reduce the runtime of tasks and the energy consumed by the devices. MEC servers do not have the same capabilities as CS servers, and because of this, the utilization of ESs is sensitive. The control-based offloading of tasks provides real-time decision-making[24].

Accordingly, hybrid algorithm 1 is proposed for load balancing when the payload is generated by devices that must be transmitted to the virtual edge clusters.

Algorithm 1. Hybrid load balancing algorithm for edge orchestration.

S1 = Situation 1 (non-critical data)
S2 = Situation 2 (critical data)
v_g = Value generated by a device
LB = Load balancer
EB = Edge broker
ES = Edge server
EC1 = Edge cluster 1 of ESs with weights
EC2 = Edge cluster 2 of ESs with weights
CS = Cloud server
LC = Least connection algorithm
WLC = Weighted least connection algorithm
Initialize LB {
  assign device priorities
  assign weight to ECs and ESs
  assign calculators for connections in ESs
}
Loop() {
  check device priority {device with higher priority first}
  for each v_g do
    check the size && type
    if v_g = S1 then
      publish v_g to the EB

```
    use the LC algorithm to assign request to EC1
    check number of current connections
      ES with preferred capabilities in EC1 subscribes to obtain v_g
  else if v_g = S2, then
    publish v_g to the EB
    use the WLC algorithm to assign the request to EC2
    check capacity weights
    select ESs
    check the number of current connections in each ES
    ES with least connections in EC2 subscribes to obtain v_g
  else
    assign request to CS
    CS subscribes to EB to get v_g for processing
  end if
  end for
}
```

Hybrid algorithm 1 provides load prioritization and payload control to meet the desired QoS requirements of IIoT networks and the best performance of the application running. The random algorithm would also be a good solution for critical resources because it provides load distribution, but only when nodes have the same specifications.

4 IIoT communication protocols

As the IIoT devices have been ready to produce the data, the next step is how to transmit the different kinds of data produced by different kinds of devices. To solve such interoperability and transmission issues, we first performed a detailed comparative analysis of the most suitable transmission protocols for the IIoT era, as shown in Table 2[1,25].

Various frameworks, such as the Ponte IoT framework (https://github.com/eclipse/ponte) and Atlas framework proposed by Khaled et al.[26], provide an efficient and interoperable solution for communication with different protocols at the edge of a network. Such frameworks can eliminate interoperation issues by acting as a gateway at the edge of a network. Ponte and Atlas support the CoAP, MQTT, and HTTP protocols, with the latter being advantageous in terms of energy consumption[27,28].

5 Proposed digital twin architecture

In relation to the design of the proposed deterministic multi-hop wireless sensor network, the position of nodes is fixed, resulting in a simple control and implementation of the system. However, in many cases, node locations are unknown. Therefore, nodes must operate in a dynamic and distributed manner, which provides greater flexibility and scalability but requires more complex algorithms for node control.

The proposed network can be characterized as an aggregating digital twin network because the nodes are of a large volume inside the industry. Each of the nodes is near the other nodes, which can cause data redundancy. With the correct collection and transmission methods in the digital twin, this can be avoided. This results in less network congestion and less energy but increased computing performance and memory. Figure 1 shows the proposed digital twin communication model with a multi-access edge-cloud framework.

The proposed IIoT communication model, as shown in Figure 1, consists of three layers. The first layer

Table 2 Comparative analysis of the most common IoT protocols

	CoAP	MQTT	XMPP	AMQP
Architecture	Client/ Server and Client/ Broker	Client/ Broker or Broker / Bridge	Client/ Server and Client/ Broker or Broker / Bridge	Client/ Server and Client/ Broker
Model	Publish/ Subscribe and Request/ Response	Publish/ Subscribe	Publish/ Subscribe	Publish/ Subscribe and Request/ Response or Broker/ Bridge
Transport Protocol	UDP, SCTP	TCP	TCP SMTP, EXI	TCP, SCTP
Header Size	4 bytes	2 bytes	–	8 bytes 64 bytes
Topic Length	–	2 bytes	–	–
Message Size	–	26 bytes	–	–
Payload	Small	256 MB	–	Small
Security	DTLS/IPSec	TLS/SSL	TLS/SSL SASL	TLS/SSL IPSec & SASL
Communication Pattern	REST based	Topic based	–	–
Encoding Format	Binary	Binary	–	Binary
License	Open Source	Open Source	Open Source	Open Source
Default Port	5683/5684	1883/8883		5671/5672
App Portability	√	√	√	√
Flexibility	Cacheability HTTP mapping	√	√	√
Lightweight	√	√	√	√
Reliability	Reliability mechanism	√	√	√
Scalability	Complex	Simple	–	–
Interoperability	√ Essential	Challenge (DM payloads)	√	√
Heterogeneity	√	√	√	√
Durability	√	√	√	√
Performance	High	Middle (Binary + TCP)	High	High (HTTP + XML)
Bandwidth	Low	Low		Low
Latency	Low	Low	Low	Low
Overhead	Low header overhead	Low	Low	Low
Complexity	Low parsing complexity	Low	Low	Low
QoS	2-tier	3-tier	–	2-tier
Energy	Low	Medium to High	–	Low

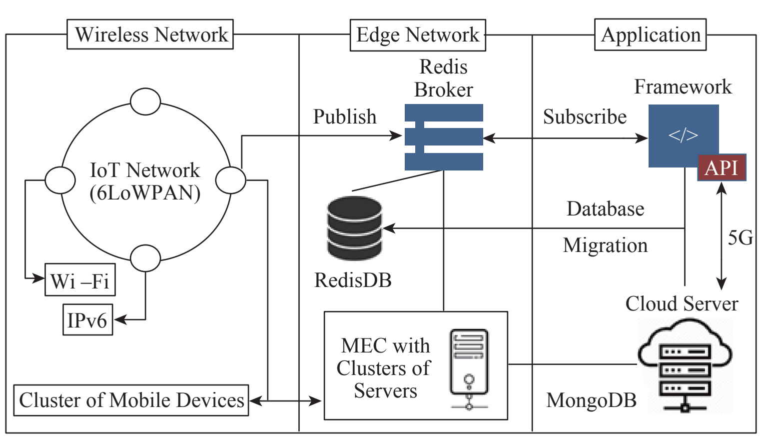

Figure 1 Proposed IIoT communication model.

constitutes the physical, connectivity, or network and abstraction sublayers. Specifically, in the physical sublayer, various devices are installed. In the connectivity sublayer, the devices are connected in a way and topology so that they can produce and transfer data over time. In the abstraction sublayer, the differences between the valuable information and the total data generated, have been reduced. The second layer comprises ESs, a database, and a broker. Specifically, the valuable data produced are forwarded through the broker device to the ESs for a good real-time manipulation and management and finally stored for a specific period of time inside the local database. The third layer consists of an application, a CS, and a cloud database. Specifically, the application updates data stored in the local database in real time efficiently, and cloud services are provided for complex analytics and long-term storage.

The Contiki OS is used with its useful applications for the digital twin network design and testing. Several emulations in the Cooja emulator are run to measure in a single cluster the network performance, energy consumption, packet loss, latency, routing metrics, protocol performance, and additional measurements, which is discussed later in this section. In Figure 2, a network cluster of mobile devices that are implemented and run for 30min with Cooja can be observed.

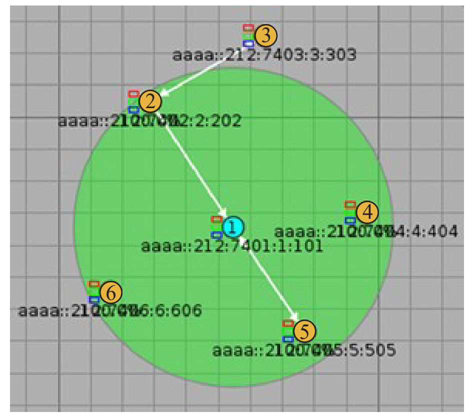

Figure 2 Cluster of devices with the Cooja emulator.

The specifications of specific clusters in the network are presented in Table 3.

Table 3 Specifications of the specific cluster

Components	Specifications
Operating System/Simulator/Emulator	Contiki OS 3.0 / Cooja Emulator
Radio Medium	Unit Disk Graph Medium (UDGM): Distance Loss
OperatingFrequency	2.9 GHz
Data Rate	250 kbps
Protocols	6LoWPAN, IEEE 802.15.4, RPL, CoAP
# of nodes	6
Tx/Rx	50 m × 50 m
Runtime	1800s
Packet Size	127 bytes
PHY and MAC Protocol	IEEE 802.15.4 and CSMA/CA
Type of Mote	Sky Mote

Figure 3 depicts the battery lives of the six nodes. This is one of the most important aspects that must be considered and tested during IoT development. The dissatisfaction of users, decreased battery life, and increased cost are some of the results of avoiding measuring the energy consumption of a system, platform, or application.

For the energy consumption estimation of an application, the measurements could be held with specific software, but in the IoT platform, the complexity is high.

Figure 4 displays the historical power consumption in mW per second for six nodes (**1.1**, **2.2**, **3.3**, **4.4**, **5.5**, and **6.6**). Figures 5 and 6 show the instantaneous and average power consumption for the six nodes, respectively. Both figures show the energy consumed by the low power mode, CPU, radio listen, and radio transmission.

The energy consumption of each node was calculated by applying Equation 1:

$$PCON = (EV \times Curr \times V) / (RTsec \times Runtime), \quad (1)$$

where *PCON* is the power consumption, *EV* is the energest value, *Curr* is the current, *V* is the voltage, *RTsec* is the timer in seconds, and *Runtime* is the time in which the simulation runs.

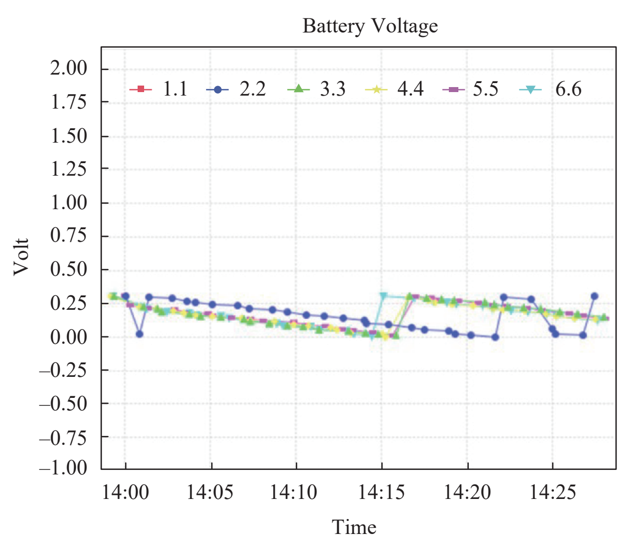

Figure 3 Battery voltage for 30 min of continuous data exchange for six nodes (1.1, 2.2, 3.3, 4.4, 5.5, and 6.6).

A low-power and lossy network typically uses

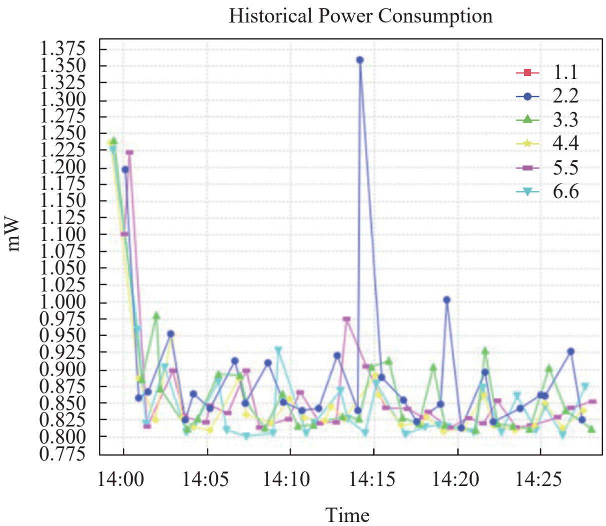

Figure 4 Historical power consumption in mW per second.

battery-constrained motes. To measure the energy consumption of the network, the time the mote is ON can be divided into time intervals. This process results in the average radio duty cycle of the network, which is presented in Figure 7 and can be described as Equation 2:

$$ARDC = Ton / Tint, \quad (2)$$

where *Ton* is the time the mote is ON and *Tint* is the time interval.

Figures 8 and 9 display the estimated time of transmission (ETX to the next hop) and the received packets per time for the six nodes (**1.1**, **2.2**, **3.3**, **4.4**, **5.5**, and **6.6**), respectively.

To estimate the network reliability, Equation 3 is used, which expresses the packet delivery ratio

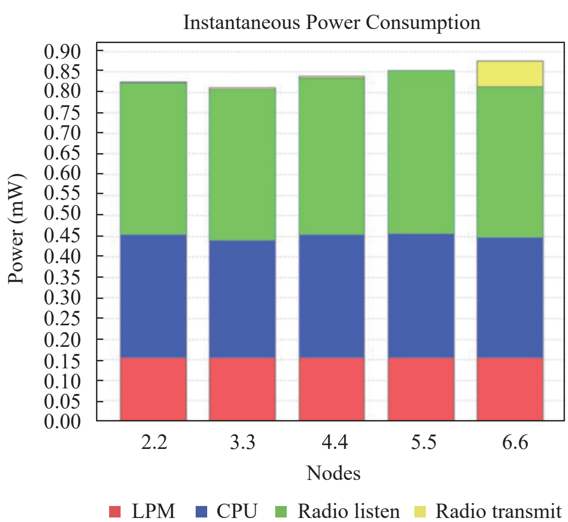

Figure 5 Instantaneous power consumption.

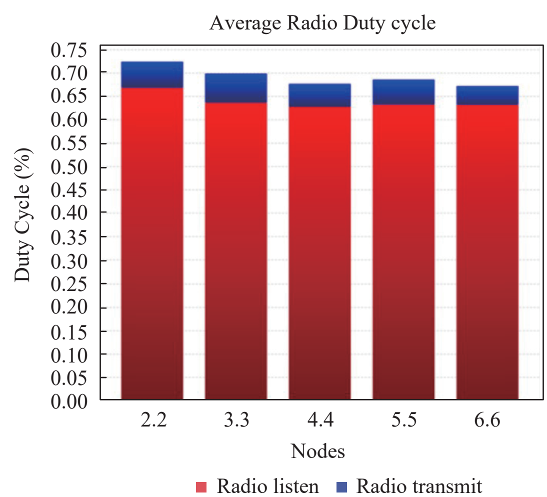

Figure 7 Average radio duty cycle.

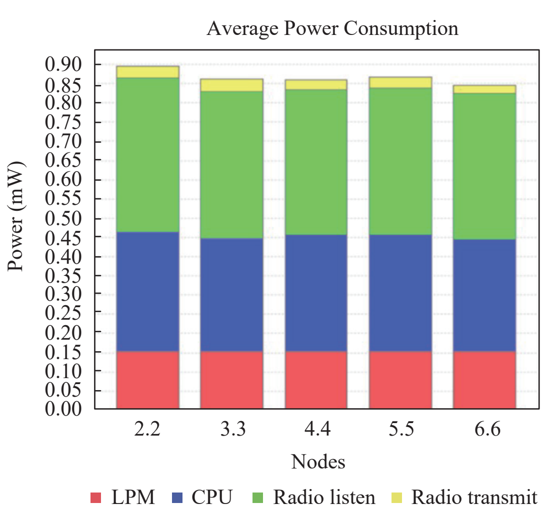

Figure 6 Average power consumption of each node.

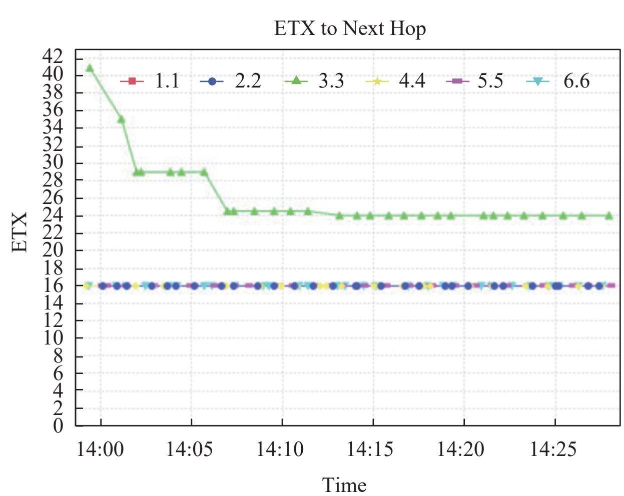

Figure 8 Estimated number of transmissions (ETX to the next hop).

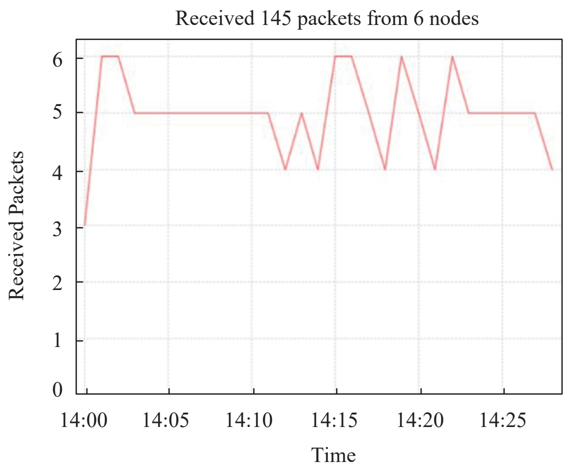

Figure 9 Received packets per minute.

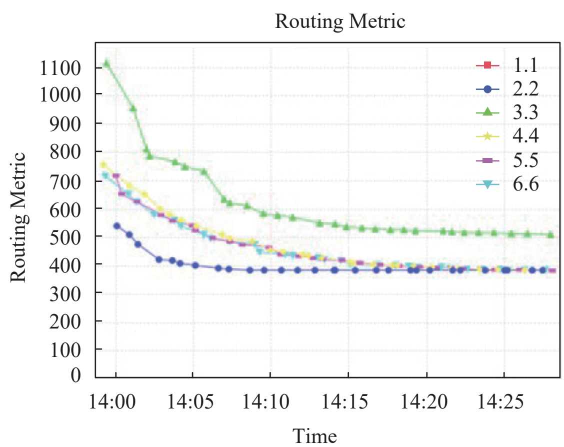

Figure 10 Received packets per minute.

(PDR).

PDR = Packets Received / Packets Transmitted (3)

Figure 10 presents the received packets per minute for the six nodes (**1.1**, **2.2**, **3.3**, **4.4**, **5.5**, and **6.6**). In the middle layer of the proposed framework, a combination of two nonrelational databases (NoSQL) plays a key role in the entire system because it provides flexibility and scalability in BD real-time applications. Such databases are MongoDB and Redis, and they can be observed in Figures 1 and 11.

Redis acts as a broker that delivers messages; thus, it has been used in the edge-fog layer for real-time communications. If needed, it can also serve in the cloud. In addition, it has been used as a local database and cache storage. It serves the processing of data even in heavy situations in a few milliseconds. MongoDB has also been studied and configured inside the framework as a database for a specific volume of data and to efficiently handle large amounts of data produced by IoT devices. It also provides cloud service. Table 4 presents a comparative analysis of the most common databases.

Figure 11 shows the digital twin system model and flow. Specifically, the devices are divided into clusters depending on the feasibility of each device, critical and non-critical resources, and thus the priority of the devices. The flow of data generated by these clusters and the different protocols used can also be observed. Furthermore, the proposed middleware is shown next, which consists of clusters of two ESs that can manage the loads using suitable algorithms depending on each occasion that provide real-time results (analytics) to end mobile users. In addition, this layer consists of broker devices, which can efficiently handle the communication and publication of data and the conjunction of the two aforementioned databases. Then, a CS can access the data by subscribing to the ESs or migrating data from the database system. Finally, the CS responds to the results from a deep analysis of the data[29].

An API was developed with an open-source PHP framework, which provides a document object model for the API. In simple words, it contains a structure for efficient application development and is based on the MVC architecture[20,21]. An application that runs on a mobile device was developed by combining the Laravel and VueJS frameworks. These frameworks provide

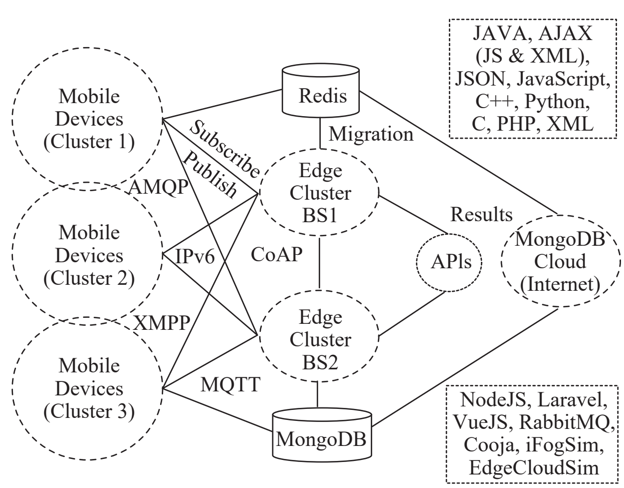

Figure 11 Proposed IIoT multi-access edge framework architecture.

Table 4 Comparative analysis of the proposed framework among others

	[1]	[17]	[19]	[29]	Proposed
MEC	–			–	
FC	–	–		–	
Cloud-assisted					
IIoT				–	
Network	5G	5G	5G	6G	5G
ML		–	–	–	–
DL		–	–	–	–
Performance	High	–	High	High	High
Security					
Energy efficiency		–	–	High	High
Latency	–	Low	Low	Low	Low
Data rate	–	High	High	High	High
Overhead	–	–	–	–	Low
Transmission Speed	–	–	High 1s in fog and 16s in edge	High (3–5s)	High
Interoperability					
Flexibility					
Adaptability				–	
Scalability			–	–	
Reliability	High	–	–		High
Broker	–	–		–	
Mobility	–				
Caching	–	–	–		
Algorithms for caching	–	–	–	LFRU (LRU + LFU)	Hybrid (WRR & WLC)
Task offloading	–	–	–	–	
Load balance	–	–	–	–	
Complexity	Medium	–	–	Low	Low

authentication for the user and therefore for the device.

6 Evaluation, testing, and experimental results

To begin with the evaluation of the proposed scenario, the open-source EdgeCloudSim simulator was configured and used to simulate the MEC scenario for several tasks and nodes simultaneously, as inside an industry. Multi-tier with edge orchestrator scenarios, which require many servers (edge and cloud), were implemented. This simulator used five memory management algorithms, namely, Random_Fit, Worst_Fit, Best_Fit, First_Fit, and Next_Fit, which were tested.

Random_Fit allocates a random memory block from a group of chosen blocks that are tracked. This algorithm has complex implementation. Worst_Fit allocates to the largest partition one process, but if another large one arrives, it cannot be allocated. Best_Fit allocates the smallest partition that the process arrives at. First_Fit introduces internal and external fragmentation issues.

The first is the allocation of memory slices at the starting point of memory, whereas the second is caused by the slicing of partitions while looking for an empty one. Thus, a big process that arrives is dropped. Next_Fit, which is an extension of First_-Fit, begins the next search from the last point.

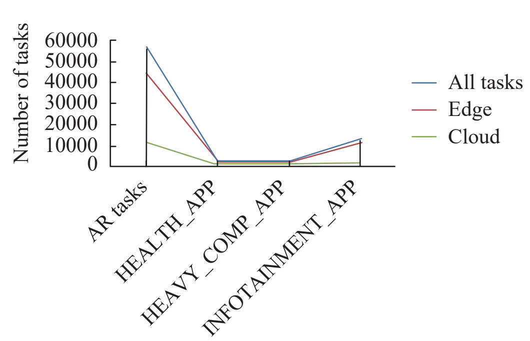

Figure 12 Number of tasks in different application domains.

Figures 12 and 13 present the number of tasks

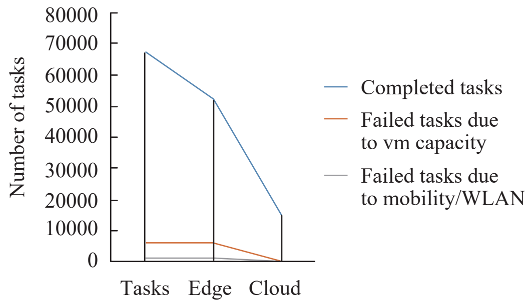

Figure 13 Number of completed and failed tasks in edge and cloud.

completed in four different application domains and the number of completed and failed tasks on the edge and cloud, respectively.

In this section, the results of the tests performed to evaluate the proposed digital twin architecture are provided. Table 4 presents a comparative analysis among the various implementations and the proposed one.

Table 4 reveals that the proposed framework satisfies the design requirements[22]. Specifically, QoS parameters, such as low latency, bandwidth, data loss ratio, jitter, payload prioritization, and load control, were analyzed. In addition, owing to its real-time capabilities, the proposed network can handle multiple critical and noncritical applications with no interference.

The network devices and ESs were grouped into clusters. These clusters are capable of handling various devices, ensuring the segmentation of the proposed network. Moreover, reliability was achieved because the load was delivered efficiently and in the expected order.

Because of the use of the load balancing algorithm, the convergence speed and system utilization were at high levels, processing time for training was lower because of the ES with a low load selection, competition of motes was reduced, and average delay among the users was reduced. Confidentiality and integrity were guaranteed because of the authentication of each device and user. The proposed network can easily be configured again to integrate new devices if required.

The complexity of the proposed algorithm is of high importance because a wireless, low-power, and lossy network consists of constrained devices in terms of energy consumption, and different layers in the digital twin network use different components, making it even more critical. Capacity, load, and energy consumption have been considered in order to measure the complexity of the algorithm. In the proposed system, complexity was maintained at low levels.

7 Conclusions

In this study, the novel technology of MEC was studied and combined with the digital twin technology. MEC provides scalability, reliability, security, and efficient data control and storage. The proposed algorithm provides efficient data delivery and task offloading. The results show that processing, security, complexity, control, energy consumption, and reliability are enhanced. In addition, the proposed framework and application provide authentication and integrity to end users and IoT devices.

Declaration of competing interest

We declare that we have no conflict of interest.

References

1 Hsiao C H, Lee W P. OPIIoT: design and implementation of an open communication protocol platform for industrial Internet of Things. Internet of Things, 2021, 16: 100441
DOI: 10.1016/j.iot.2021.100441

2 Mahalingam M. Challenges in industrial internet of things. International Journal of Exclusive Global Research, 2021, 6(8)

3 Gunasekaran A, Ngai E W T. Information systems in supply chain integration and management. European Journal of Operational Research, 2004, 159(2): 269–295
DOI: 10.1016/j.ejor.2003.08.016

4 Pydi H, Iyer G N. Analytical review and study on load balancing in edge computing platform. In: 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC). Erode, India, IEEE, 2020, 180–187
DOI: 10.1109/iccmc48092.2020.iccmc-00036

5 Husain S, Kunz A, Prasad A, Samdanis K, Song J. Mobile edge computing with network resource slicing for Internet-of-Things. In: 2018 IEEE 4th World Forum on Internet of Things. Singapore, IEEE, 2018, 1–6
DOI: 10.1109/wf-iot.2018.8355232

6 Choi T M, Wallace S W, Wang Y. Big data analytics in operations management. Production and Operations Management, 2018, 27(10): 1868–1883
DOI: 10.1111/poms.12838

7 Zhou H, She C Y, Deng Y S, Dohler M, Nallanathan A. Machine learning for massive industrial Internet of Things. 2021

8 Wang D, Chen D, Song B, Guizani N, Yu X, Du X. From IoT to 5G I-IoT: the next generation IoT-based intelligent algorithms and 5G technologies. IEEE Communications Magazine, 2018, 56(10): 114–120
DOI: 10.1109/mcom.2018.1701310

9 Fuller A, Fan Z, Day C, Barlow C. Digital twin: enabling technologies, challenges and open research. IEEE Access, 2020, 8: 108952–108971
DOI: 10.1109/access.2020.2998358

10 VanDerHorn E, Mahadevan S. Digital Twin: generalization, characterization and implementation. Decision Support Systems, 2021, 145: 113524
DOI: 10.1016/j.dss.2021.113524

11 He B, Bai K J. Digital twin-based sustainable intelligent manufacturing: a review. Advances in Manufacturing, 2021, 9(1): 1–21
DOI: 10.1007/s40436-020-00302-5

12 He R, Chen G, Dong C, Sun S, Shen X. Data-driven digital twin technology for optimized control in process systems. ISA Transactions, 2019, 95: 221–234
DOI: 10.1016/j.isatra.2019.05.011

13 Qi Q, Tao F, Hu T, Anwer N, Liu A, Wei Y, Wang L, Nee A Y C. Enabling technologies and tools for digital twin. Journal of Manufacturing Systems, 2021, 58: 3–21
DOI: 10.1016/j.jmsy.2019.10.001

14 Madni A, Madni C, Lucero S. Leveraging digital twin technology in model-based systems engineering. Systems, 2019, 7(1): 7
DOI: 10.3390/systems7010007

15 Stefan B, Christoph H, Rosen R. Next generation Digital Twin. 2018

16 Dutta P, Choi T M, Somani S, Butala R. Blockchain technology in supply chain operations: applications, challenges and research opportunities. Transportation Research Part E: Logistics and Transportation Review, 2020, 142: 102067
DOI: 10.1016/j.tre.2020.102067

17 Goudos S K, Dallas P I, Chatziefthymiou S, Kyriazakos S. A survey of IoT key enabling and future technologies: 5G, mobile IoT, sematic web and applications. Wireless Personal Communications, 2017, 97(2): 1645–1675
DOI: 10.1007/s11277-017-4647-8

18 Islam A, Debnath A, Ghose M, Chakraborty S. A survey on task offloading in multi-access edge computing. Journal of Systems Architecture, 2021, 118: 102225
DOI: 10.1016/j.sysarc.2021.102225

19 Borsatti D, Davoli G, Cerroni W, Raffaelli C. Enabling industrial IoT as a service with multi-access edge computing. IEEE Communications Magazine, 2021, 59(8): 21–27
DOI: 10.1109/mcom.001.2100006

20 Liu K H, Liao W. Intelligent offloading for multi-access edge computing: a new actor-critic approach. In: ICC 2020-2020 IEEE International Conference on Communications. Dublin, Ireland, IEEE, 2020, 1–6
DOI: 10.1109/icc40277.2020.9149387

21 Zhao Z, Zhao R, Xia J, Lei X, Li D, Yuen C, Fan L. A novel framework of three-hierarchical offloading optimization for MEC in industrial IoT networks. IEEE Transactions on Industrial Informatics, 2020, 16(8): 5424–5434
DOI: 10.1109/tii.2019.2949348

22 Yu J J, Zhao M X, Li W T, Liu D, Yao S W, Feng W. Joint offloading and resource allocation for time-sensitive multi-access edge computing network. In: 2020 IEEE Wireless Communications and Networking Conference. Seoul, Korea (South), IEEE, 2020, 1–6
DOI: 10.1109/wcnc45663.2020.9120620

23 Mao Y, You C, Zhang J, Huang K, Letaief K B. A survey on mobile edge computing: the communication perspective. IEEE Communications Surveys & Tutorials, 2017, 19(4): 2322–2358
DOI: 10.1109/comst.2017.2745201

24 Lee C K M, Huo Y Z, Zhang S Z, Ng K K H. Design of a smart manufacturing system with the application of multi-access edge computing and blockchain technology. IEEE Access, 2020, 82: 8659–28667
DOI: 10.1109/access.2020.2972284

25 Jaloudi S. Communication protocols of an industrial Internet of Things environment: a comparative study. Future Internet, 2019, 11(3): 66
DOI: 10.3390/fi11030066

26 Khaled A E, Helal S. Interoperable communication framework for bridging RESTful and topic-based communication in IoT. Future Generation Computer Systems, 2019, 92: 628–643
DOI: 10.1016/j.future.2017.12.042

27 Martin D, Kühl N, Schwenk M. Towards a reference architecture for future industrial Internet of Things networks. In: 2021 IEEE 23rd Conference on Business Informatics. Bolzano, Italy, IEEE, 2021, 1–9
DOI: 10.1109/cbi52690.2021.10049

28 Mardani A, Saraji M K, Mishra A R, Rani P. A novel extended approach under hesitant fuzzy sets to design a framework for assessing the key challenges of digital health interventions adoption during the COVID-19 outbreak. Applied Soft Computing, 2020, 96: 106613
DOI: 10.1016/j.asoc.2020.106613

29 Stergiou C L, Psannis K E, Gupta B B. IoT-based big data secure management in the fog over a 6G wireless network. IEEE Internet of Things Journal, 2021, 8(7): 5164–5171
DOI: 10.1109/jiot.2020.3033131