Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2020 SCPE. Volume 21, Issues 3, pp. 463–477, DOI 10.12694:/scpe.v21i3.1746

A PARALLELIZATION BASED DATA MANAGEMENT FRAMEWORK FOR PERVASIVE IOT APPLICATIONS

SANIYA ZAHOOR, ROOHIE NAAZ MIR*

Abstract. Pervasive Internet of Things (IoT) is a research paradigm that has attracted considerable attention nowadays. The main aim of pervasive IoT is that in the future, the everyday objects (devices) would be accessible, sensed, and interconnected inside the global structure of the Internet. But in most of the pervasive IoT applications, the resources of an IoT device such as storage, processing, and energy are limited; as such there is a need for management of resources in such applications. Multiple aspects related to the data such as the type of data, size of data, number of transmission and reception of data packets, the structure of data, etc are taken into consideration while managing the resources of pervasive IoT applications. Therefore data management is essential for the management of limited resources in such applications. This paper presents the recent studies and related information in data management for pervasive IoT applications having limited resources. This paper also proposes a parallelization based data management framework for resource-constrained pervasive applications of IoT. The comparison of the proposed framework is done with the sequential approach through simulations and empirical data analysis. The results show an improvement in energy, processing, and storage requirements for the processing of data on the IoT device in the proposed framework as compared to the sequential approach.

Key words: Internet of Things, Resource-constraints, Security, Tiny encryption, Data aggregation, Parallelization

AMS subject classifications. 68M11

1. Introduction.

Pervasive Internet of Things is the evolution of the Internet designed to sense, collect, analyze, and distribute the data via smart, programmable, light-weight, miniaturized IoT devices (nodes). The main goal of pervasive IoT is to form the network of day-to-day life objects and make them programmable using wireless and sensor technologies [1]. The pervasiveness of IoT eases everyday activities such as data exchange by devices while sensing and reacting to events. There are numerous applications of pervasive IoT that come from different sectors such as healthcare, agriculture, homes, offices, waste management, transport, weather monitoring, water supply, etc. But in most of these applications, resources such as storage, processing, and energy are constrained [4]. These resource constraints impact the software design at various levels e.g., the lack of sufficient hardware features facilities the design of small memory footprints. The hardware constraints also impact the design of many protocols and algorithms e.g., data aggregation algorithms executed in resource-constrained IoT networks. It also prohibits the integration of many desirable components such as GPS receiver on the IoT devices.

Due to the resource-constrained nature of pervasive applications, there are restrictions on the type of data processing algorithms that run on an IoT device. Various lightweight data processing algorithms are used on the IoT node as the feasible solution in such scenarios [3]. But as the size of data increases, even implementing a lightweight algorithm on IoT node adds resource overheads. Therefore, the data is an essential factor that imposes restrictions on the use of resources in IoT nodes of pervasive applications [4]. Multiple aspects related data are taken into consideration while managing resources of an IoT device and these include the size of data, type of data, the structure of data, number of transmission and reception of data packets, aggregation of data, duration of data storage, etc. Therefore, data management becomes necessary to manage resources in these environments.

There have been recent studies in data management solutions for the management of resources in such applications through data aggregation mechanisms [5], data storage solutions [6], virtualization techniques [7], architecture-based solutions [8], lightweight data security approaches [9], and data parallelization approaches

*Department of Computer Science Engineering, National Institute of Technology Srinagar, India (saniyazahoor@nitsri.net).

[10]. However, most of the work has been reported in Wireless Sensor Networks (WSNs) and lacks comprehensive experimental evaluations. There has been little work in Internet of Things, as such efforts are required to design a data management solution for pervasive IoT. This paper proposes a parallelization based data management framework to minimize resource overheads on the individual IoT nodes in pervasive IoT applications. The comparative analysis of resource consumption is drawn at the node level for the proposed framework with respect to the sequential approach [11] through simulations and empirical data analysis.

The rest of the paper is organized as: Section 2 presents the literature survey, Section 3 discusses proposed data management framework, Section 4 gives experimental and evaluation details, Section 5 presents the results and discussions, and section 6 gives conclusions.

2. Literature Survey. Due to the resource-constrained nature of IoT devices, pervasive IoT faces great challenges of data management in terms of storing, processing, and communicating the data [4]. Traditionally, data management is related to the data lifecycle requirement of a system. In the context of IoT, data management serves as a layer between the devices generating the data and the applications utilizing the data for analysis [12]. In our context, data management in pervasive IoT is visualized as the deployed algorithm, protocol, architecture, or framework that primarily focuses on managing limited resources of IoT devices through the proper management of data. There has been significant research in data management for pervasive applications through data aggregation mechanisms, data storage solutions, data management models, virtualization techniques, architecture-based solutions, lightweight data security approaches, and data parallelization approaches as discussed below.

2.1. Data Aggregation Mechanisms. One of the widely used and recognized techniques for data management in most of the pervasive IoT applications is data aggregation. Few comparative studies have been carried out in data aggregation for optimization of resources such as network lifetime, processing, storage, data redundancy, reliability, latency, etc [13]. The possible mechanisms for data aggregation include centralized, cluster-based, and tree-based. In centralized data aggregation, a central node aggregates the data from each IoT node [14]. The work in [15] presents a centralized data aggregation approach that addresses scalability and heterogeneity issues but the limitation of the work is the single point of failure. In cluster-based data aggregation, a cluster head is selected in each cluster to reduce the data traffic in the network. Considerable work in this includes Chinese remainder based theorem [16], cross-layer data aggregation [17], low resource cost data aggregation [18], etc. In tree-based data aggregation, the intermediate nodes perform the aggregation of data. Considerable work in this includes energy-efficient tree approach [19][20], tree-based data aggregation approach to balance the energy and network load [21], etc. However, most of the work in IoT data management through data aggregation has been reported in WSNs and lacks a comprehensive performance evaluation framework.

2.2. Data Storage Solutions. In pervasive IoT applications, IoT devices generate data rapidly, as such, it becomes necessary to store the data efficiently. Further, the heterogeneity of IoT data collected from different sources necessitates the need for efficient data storage solutions ready to deal with heterogeneous IoT data. To address such problems, various data storage solutions have been proposed that allow efficient storage ad integration of heterogeneous data viz., structured, and unstructured data [6]. Several aspects are taken into consideration for data storage solutions and these include the type of data, location of data, duration of storage, etc. Several data storage schemes that help in management of resources include data-centric storage [22], provenance-aware storage [23], real-time databases [24], centralized storage [25][26], etc.

2.3. Architecture based solutions. Depending on the specific requirements and design constraints of applications, the architecture of IoT varies from one application to another. Based on the available study, IoT architectures have been classified into centralized, distributed, and service-oriented architectures. The centralized architectures are one of the most widespread models for data management in IoT applications having limited resources. In this, the sensed data from the IoT devices are transferred to a single central device (or location), which combines, processes and presents the information to the end-users. The work in [7] presents a centralized scheduling method, 6TiScH in which the amount of resource consumption is modeled using historical information about resource consumption. The work in [27] also presents a centralized architecture, based on web resources, to decouple the domain of heterogeneous devices from the application development.

But these approaches fail to guarantee scalability and interoperability among devices of different application domains.

To overcome the disadvantages of centralized architectures, the distributive IoT architectures have been introduced. The distributive IoT architectures provide services at the node level and at the network level by the collaboration of nodes and users to achieve a common goal. One of the disadvantages encountered in this approach is a security breach that occurs due to device mobility and network heterogeneity [28]. To overcome this in pervasive and distributive IoT, few studies introduce master key for entity authentication [29] [30] [31]. Another disadvantage of distributive IoT architecture is excessive energy consumption and to overcome this, the work in [32] presents a decentralized scheduling approach based on Proportional, Integral, and Derivation algorithm that operates dynamically and controls the data traffic in the network to reduce the unnecessary consumption of energy.

The Service-Oriented architecture (SOA) is a distributive architecture of autonomous services executing on nodes with different service providers. SOA enables the decomposition of large networks into well-defined IoT networks in which nodes execute an arbitrary number of services and exchange information among them without human intervention [33]. Considerable work has been carried out in Service-Oriented Architectures for WSN and IoT applications. These include OASIS Reference Model for Service Oriented Architecture (SOA-RM) [34], a Web Service Middleware for Ambient Intelligence (aWESoME) [35], Knowledge-Aware and Service-Oriented architecture (KASO) [36], micro-subscription management system (mSMS) [37] , etc. SOA is considered applicable for IoT environments but there are many challenges such as security, limited resources, etc that need to be taken care of while adopting SOA into the IoT environment [38].

2.4. Virtualization Techniques. IoT environment consists of a huge number of IoT devices that produce data of variety in type, size, and formation, and this imposes great challenge of data management in pervasive applications. Virtualization techniques can efficiently handle this complexity in data. In pervasive IoT, the trend is to introduce virtualized environments at the node level or the network level. At the node level, multiple applications run their tasks concurrently on an individual IoT node Considerable work in this includes the container-based virtualization used on IoT devices for creation and initiation of virtualized instances [39], docker containerization used on Edge platforms to improve the manageability of resources and services [40], light-weight virtualization for IoT gateways that provides better IoT services [41], container Edge-cloud PaaS architecture that minimizes the consumption of energy [42], etc. However, it considers only a limited number of Raspberry Pi boards for performance evaluation and lacks comprehensive energy and power evaluations. On the other hand, the network-level virtualization creates a Virtual Sensor Network (VSN) containing a subset of sensor nodes that performs a given task, while other sensor nodes remain reserved for other tasks e.g. the Radio Access Network (RAN) virtualization that dynamically provides an isolated network in IoT [43], Long-Term Evolution (LTE) network virtualization using hypervisor software [44], etc.

2.5. Lightweight Data Security Approaches. Enabling data security in pervasive IoT is a challenging task due to the resource limitations [45]. Several data management strategies are used for optimizing the performance of security algorithms at hardware and software levels [46][47], but only a few are implemented in resource-constrained IoT networks. Recently, there has been a huge demand for lightweight authentication and encryption for securing resource-constrained IoT devices. These lightweight algorithms aim to balance security and resource costs to achieve privacy and performance advantages in IoT [48][49][50]. Due to their low processing power, limited battery life, small size, and small memory, the lightweight data security algorithms are considered efficient for securing resource-constrained devices in pervasive IoT applications [51][52][53].

Several lightweight two-factor user authentication schemes have been proposed for WSNs, but such schemes seem vulnerable to several attacks such as replay, denial of service, etc. Few lightweight security schemes, based on the computation of hashing function, have also been proposed for WSNs [54][55] but the work is vulnerable to several attacks (e.g., login identity attack). Various schemes such as password-based user authentication, mutual authentication, etc have also been proposed but these fail to satisfy mutual authentication between the base station and the sensor node [56]. To address such issues, the work in [57] proposes a lightweight protocol to secure IoT devices via portal controllers; the proposed protocol preserves the privacy of communications; however, it generates an overhead regarding the number of messages exchanged among IoT devices.

Predominant research is done on lightweight key management protocols for IoT devices that guarantee data confidentiality and constrained node authentication during data transmission along the channel; the limitation of the work is that the security protocol does not specify resource overheads in IoT environments [58]. The work in [59] proposes a similar lightweight security framework for resource-constrained smart objects but the proposed framework was not integrated into the resource-constrained IoT environments to evaluate its suitability. The work in [60] presents a lightweight authentication protocol to enable security on computationally constrained RFID tags; the proposed protocol guarantees minimum computation overhead with better authentication among RFID tags. To address the security and privacy concerns in constrained IoT environments, SecKit, a security toolkit has been proposed [61]; the drawbacks of this approach is it does not provide information on how to deploy security and privacy solutions for devices operating in a dynamic IoT environment.

There has been also tremendous research on lightweight data security schemes based on attribute encryption [62][63][64]. The work in [62] proposes a lightweight Attribute-Based Encryption scheme that decreases the resource overhead in terms of computation and communication of IoT environments. The work in [63] presents a similar attribute-based encryption scheme that ensures a trade-off between computation and storage capacity of constrained devices. The work in [64] discusses a similar lightweight attribute-based encryption scheme for heterogeneous IoT applications with a disadvantage of high bandwidth consumption. Therefore, security in such environments is a serious issue because of resource constraints in IoT devices and networks; also the number of attacks is a bit higher [65]. The existing security techniques do provide a basis for privacy and security, but these techniques cannot be used in resource-constrained IoT without modifications [66].

2.6. Data Parallelization Approaches. There has been considerable research in data parallelization for efficient management of limited resources. Several serial approaches are being used for analyzing small data sets but it results in increased resource overheads especially storage and processing [10]. To address this, data parallelization approaches are being used that harness multiple processing units to solve this problem wherein multiple processing units execute the computation simultaneously to reduce computing time [67].

In IoT applications, which follow in-network processing IoT architectures, the individual IoT nodes are overloaded due to the size of data [68]. To address this, parallel approaches are being widely used in many IoT applications such as medical imaging, bio-informatics, graph mining, etc. The work in [69] presents a parallel computing framework to integrate diverse computing resources for manufacturing IoT applications. The work in [70] proposes a similar framework that uses the cloud to optimize process planning. The work in [71] develops a parallel algorithm to achieve highly efficient service decomposition and optimal selection. However, most of these studies focus on strategical problems in manufacturing e.g., job scheduling and service optimization. In pervasive IoT, parallel algorithms are mapped on FPGA to obtain minimal power consumption [72][73], while others use data parallelization to reduce the device overload [74]. However, most of the work on improving the resources of IoT devices is done at the hardware level only.

Table 2.1 presents the classification of recent studies and related information in data management for pervasive IoT.

3. Proposed Data Management Framework. In pervasive applications of IoT, the resource utilization should be optimized because of resource constraints, as such, the algorithms and protocols used in the data management framework should be designed accordingly. In resource-constrained nodes, there are restrictions on the use of data processing algorithms due to resource overheads. To reduce this overloading on IoT nodes, we propose a data management framework that implements any lightweight algorithm with low resource overheads. The framework uses the concept of data parallelization on multiple nodes for minimizing the overloading on the individual IoT node in IoT networks. For the implementation of a lightweight algorithm in such a scenario, the Tiny Encryption Algorithm is used. The TEA is a lightweight cryptographic algorithm and due to its simplicity of description and implementation, the algorithm is more suited for the resource-constrained IoT scenario.

Since, for the implementation and evaluation purposes, we have used a data security algorithm to test the proposed framework, so the prime objectives of the proposed framework shall be security along with optimal resource utilization. Table 3.1 highlights the various resource and security concerns of various pervasive IoT applications.

3.1. Formulation of Proposed Framework. The proposed data management framework applies to homogeneous in-network processing architectures of IoT which includes distributed data-centric IoT architec-

Table 2.1
Classification of Recent Studies and Related Information in Data Management for Pervasive IoT

IoT Data Management	**Related Work**	**Performance Metrics**	**Issues**	**Research Possibilities**
Data Aggregation Mechanisms	Centralized data aggregation [19]	Heterogeneity, Scalability	Single point of failure	With modifications manage resources in IoT
	Cluster based data aggregation [16]	Energy efficiency,Security	High Hardware costs	
	Tree based data aggregation [14]	Network lifetime, Throughput, Delay	High computational cost	
Data Storage Solutions	Data-centric storage [22], Provenance-aware storage [23], Real time databases [24], etc	Storage utilization	Energy Consumption Overhead	Limited work in IoT
Architecture based Solutions	Centralized Solutions [27]	Resource Management	Congestion and single point of failure, Scalability issues	Standardization of IoT resource management architecture
	Distributed Solutions [28]	Bandwidth utilization, Reduced number of transmissions	Security Breaches, Excessive energy consumption	
	Service Oriented Solutions [38]	Reduced number of transmissions	Security Breaches	
Virtualization Techniques	Node Level Virtualization [41, 42]	Energy, manageability of resources and services	Limited number of devices for performance evaluation	Exploration at the system level
	Network Level Virtualization [43, 44]	Dynamic network provisioning	Limited work in IoT	Exploration at the system level
Lightweight Data Security	Two-Factor User Authentication Schemes [56, 57]	Data Security	Message Exchange Overhead	Limited Work in IoT
	Key Management Protocols [58, 59]	Data Confidentiality	Limited work in resource-constrained IoT	
	Attribute-based Encyption Schemes [64, 63]	Less Computation and Communication Costs	Bandwidth Utilization	
Data Parallelization Approaches	Parallel computing frameworks [69, 70], Parallel algorithms on FPGA [72]	Less Power Consumption	Limited work at software level	More attention needed on data management at hardware level

tures, where data processing such as encryption, data fusion, etc are done at the node level. In the proposed framework, we consider IoT applications wherein the IoT nodes are resource-constrained and non-replaceable after the deployment for a particular application.

3.1.1. Assumptions. Following assumptions are taken in our IoT scenario:

- An IoT scenario consists of super-master, master and slave nodes,
- Each slave node, S_{in} must have a unique ID, I_{Miid},
- Each master node, M_i must have a unique ID, M_{iid},
- Resources of slave nodes $\{E_i, P_i, M_i\}$ is equivalent to that of master nodes $\{E_{Mi}, P_{Mi}, M_{Mi}\}$,
- Resources of super master node $\{E_{1i}, P_{1i}, M_{1i}\}$ are higher than that of slave nodes $\{E_i, P_i, M_i\}$ and master nodes $\{E_{Mi}, P_{Mi}, M_{Mi}\}$,

Table 3.1
Pervasive IoT Applications: Resource and Security Perspectives

Pervasive IoT Application	**Main Resource Concern**	**Main Security Concern**
Healthcare [75]	Limited Storage, Processing, Energy and Bandwidth	Identity of Patients, Health records of Patients, Patient-Doctor Communication, etc
Transportation and Parking [76, 77]	Limited Storage, Processing and Bandwidth	Information of driver/Traveler, Inter-vehicular communications, Breaching Traffic Lights for hijacking, etc.
Monitoring of Weather, Environments, Waste [78, 79]	Limited Storage, Processing, Energy and Bandwidth	Confidentiality of critical information, Integrity of sensed data, etc.
Agriculture [80, 81]	Limited Storage, Processing, Energy and Bandwidth	Personal details of farmers, Integrity of sensed data, False negative and False positives lead to disastrous results for crop, etc.
Offices and Homes [82]	Limited Storage, Processing and Bandwidth	Owner Authentication, Household data, Monetary loss due to privacy breaches in online purchases from home, etc.

- Distance between the slave node and the master node , *d(*M_i to I_{Mi}*)* is negligible as compared to the distance between the master node and super master *d(*E_1 to I_{Mi}*)*,
- TEA encryption is performed by slave nodes S_{in},
- Data aggregation and distribution is done by the master nodes M_i,
- Data aggregation and decryption is done by super-master node (Edge node) E_1.

3.1.2. Proposed Algorithm.

3.1.2. Proposed Algorithm. In the proposed algorithm, the data is divided into the data chunks on the master node and is sent to the slave nodes that perform the encryption of data. After encryption of data chunks is performed, the encrypted data chunks are then pushed to the super-master node that performs the decryption of data.

As shown in Figure 3.1, the proposed framework aims to parallelize TEA implementation across the virtualized OS containers of neighboring slave nodes that help in mass data parallelism. The sensed and collected data from the master node is split among N slave nodes e.g., consider a scenario with 104 blocks of 32-bit plaintext on the master node and four slave nodes are volunteering to work for the master, then each slave node gets 26 blocks of 32-bit plain text.

A slave node is selected based on availability quotient, A which is a function of resources such as energy (e) and memory (m):

$$A = f(e, m) \tag{3.1}$$

For the encryption process, the slave nodes in the cluster run TEA in their OS containers in parallel and simultaneously. After the encryption process on slave nodes is completed, the encrypted data is gathered from all slave nodes to the super master. The super master node then sends an acknowledgment to the master node on the reception of the data. As the master node receives the acknowledgment of data chunks, the cluster vanishes, and the data is deleted from the memory of the master node and slave nodes.

For the decryption process, the super master node performs the decryption of the encrypted data chunks; the resulted decrypted data chunks will be stored on separate OS containers of the super master node pertaining to the master node. The pseudocode for the proposed algorithm is given in Algorithm 1.

4. Experimental Setup. At the time of deployment, all the nodes start functioning and acquire sensing data from the physical environment. It is only at the time of data dissemination to the edge, the node requires the help of nearby nodes for the implementation of a lightweight algorithm in parallel mode. The proposed data management framework has been tested through simulations and empirical data analysis. The simulations are carried out in MATLAB and simulation parameters are set as shown in Table 4.1.

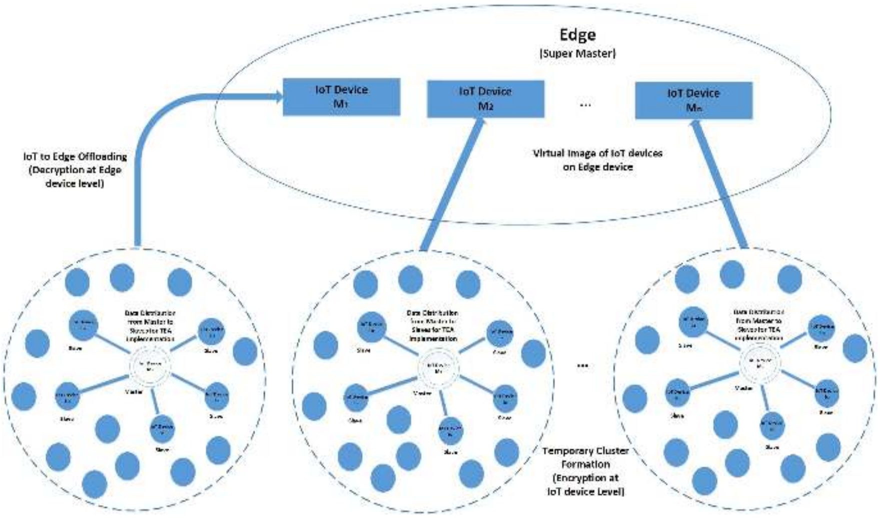
Edge
(Super Master)
IoT Device M1
IoT Device M2
...
IoT Device Mn
IoT to Edge Offloading (Decryption at Edge device level)
Virtual Image of IoT devices on Edge device
Data Distribution from Master to Slaves for TEA implementation
Slave
Master
...
Temporary Cluster Formation (Encryption at IoT device Level)

Fig. 3.1. *Proposed Data Management Framework for Resource Constrained IoT Environment*

Algorithm 1 Proposed Algorithm

```
A node, i collect data over a period of time, t
Calculate the data size periodically
As the data size of the node, i reaches a maximum size of a node D
Node i acts as a master node, $M_i$ with ID $M_{id}$
(Temporary Cluster Formation)
For each master node, Mi there is a cluster of available nearby nodes, i to act as slaves, $S_{in}$
$M_i$ broadcasts a request for cluster formation to the nearby nodes, i
N nodes accepts the request of Mi to act as slave nodes,$S_{i1}$,$S_{i2}$,$S_{i3}$,..., $S_{in}$
$S_{in}$ are selected by Mi based on Availability quotient, A where:
                    $A=f(E_i,M_i)$
Divide the data of data size, D into data chunks $D_{i1}$,$D_{i2}$,$D_{i3}$, ..., $D_{in}$
$M_i$ sends $D_{i1}$,$D_{i2}$,$D_{i3}$, ..., $D_{in}$ to the selected slave nodes, $S_{i1}$,$S_{i2}$,$S_{i3}$, ..., $S_{in}$
Each of N selected $S_{in}$ implements TEA in parallel
Each of N selected $S_{in}$ sends the encrypted data to the super master node, $E_1$
Encrypted data is collected by $E_1$ and acknowledgement is send to $M_i$
Cluster vanishes
Data is deleted on $M_i$ and $S_i$
Decryption process is performed on the $E_1$
Encrypted data Chunks from $S_{in}$ are decrypted and aggregated on the $E_1$
Data is stored on the separate containers for each master nodes ($M_i$) maintained on the super master node, $E_1$
```

Table 4.1
Simulation Parameters

Parameter	**Values**
Number of Slave IoT devices	100
Number of Master IoT devices	20
Number of super master device	1
Initial Energy of slave IoT device	300 mAh
Initial Energy of Master IoT device	300 mAh
Transmission range of a slave IoT device	40 m
Transmission range of master IoT device	40 m
Block Size	256
Key Size	128 bits

For the empirical data analysis, an Arduino based IoT device with sensors (air humidity, temperature, and pressure)is used to capture data from the physical environment. The proposed framework is tested on this data to check its suitability to map with the real-life IoT scenarios.

4.1. Performance Metrics. The comparative resource analysis is performed in terms of following metrics:

- *Round*: It is defined as the complete process which starts when the IoT node aggregates the data, implements the lightweight algorithm, and pushes the data to the edge.
- *Energy*: The energy of a node is the difference between total energy and consumed energy in an IoT system. To extend the network lifetime, it may be desirable to avoid routing through nodes with low residual energy. It is calculated as:

  $$E_i = E_t - E_c \tag{4.1}$$

  where E_i is the residual energy, E_t is the total energy of a node, and E_c is the energy consumed in one round.

  The average energy of the IoT network, E_n is calculated as:

  $$E_n = \sum E_i/i \tag{4.2}$$

  where i denotes the number of active IoT nodes.

  Let E_{TEA} and E_{P-TEA} is the average energy for TEA implementation in sequential and proposed framework respectively.
- *Storage*: Let S_r represents the residual storage on the node, then we have:

  $$S_r = S_t - S_o \tag{4.3}$$

  where S_t is the total storage of a node and S_o is the occupied space on the node. Let M_{TEA} and M_{P-TEA} is the measure of average storage requirements for TEA implementation on a single node in sequential and proposed framework respectively.
- *Processing Time*: It is the time required to process the data in a network. Processing of data involves activities such as sensing, storing, aggregation, offloading, etc. Let T_{TEA} and T_{P-TEA} is the average processing time for TEA implementation in sequential and proposed framework respectively.
- *Number of Alive Nodes versus the number of rounds*: Let A_{TEA} and A_{P-TEA} is the number of alive nodes in sequential and proposed framework respectively.
- *Degree of improvement*: Degree of improvement is referred to the percentage increase in performance of proposed framework with respect to sequential TEA implementation.

  The degree of improvement in processing time is calculated as:

  $$D_{P-TEA} = (T_{TEA}T_{P-TEA})/T_{TEA} \tag{4.4}$$

  and the degree of improvement in storage is calculated as:

  $$Dm_{P-TEA} = (M_{TEA}M_{P-TEA})/M_{TEA} \tag{4.5}$$

5. Results and Discussions. The proposed data management framework has been tested through simulations and empirical data analysis. The performance parameters used to evaluate the proposed framework with respect to the sequential approach include energy, number of alive nodes, storage, and processing time. The results are presented in the form of graphs and explanation.

Figure 5.1 represents the energy of a node during simulation (as shown by eq. 4.1 and 4.2). It is evident from the graph that the energy consumption of a node in the proposed framework is lesser as compared to the single node implementation of a lightweight algorithm. In the case of the sequential implementation of a lightweight algorithm, the energy of a node gets exhausted earlier as compared to the proposed approach. There is a 12.5% improvement in the energy of a node in the proposed framework as compared to a sequential approach.

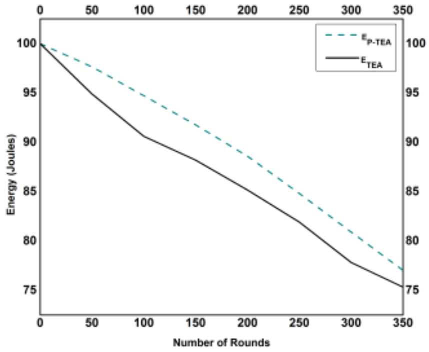

FIG. 5.1. *Energy versus Number of Rounds*

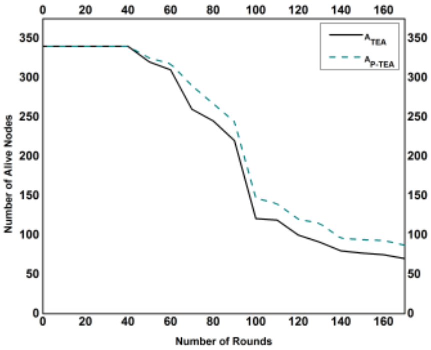

FIG. 5.2. *Number of Alive Nodes versus Number of Rounds*

Because of the parallelization of data, significant energy is saved on an IoT node. Since each node contributes to the overall network lifetime, therefore, the lifetime of the network is increased. Also, in a sequential approach, data overloading on a single node leads to the early death of the node, which can create voids in the network, and the lifetime is further reduced. Further, there is a direct relationship between energy and network lifetime. More is the energy of a node in the network, the longer is the network lifetime.

Figure 5.2 shows the number of alive nodes versus the number of rounds. It is evident from the graph that the number of alive nodes in the proposed framework is more at any particular instant as compared to the single node implementation of lightweight algorithm i.e., nodes in the proposed framework last for a longer time. Since the proposed approach balances the energy consumption and node deaths gracefully, it results in network stability. There is a 25% improvement in the number of alive nodes in the proposed framework as compared to the sequential approach.

In the proposed approach, the processing in each node is divided among different nodes and it results in fair utilization of resources in an IoT network. While in the case of a sequential approach, a particular node is overloaded while other neighboring nodes are underutilized. This results in the formation of holes in the network.

Figure 5.3 represents the comparison of storage used for the TEA implementation on a single node versus on the multiple salve nodes in the proposed framework (calculated by eq. 4.3). We observed that the storage required in sequential implementation is higher as compared to the proposed approach at a particular instant of time. In the proposed approach, as the number of slave nodes in the cluster increases, the memory requirements decrease by a factor of N i.e., higher the number of slave nodes in the cluster for the implementation of TEA

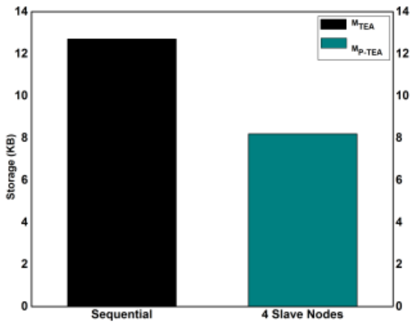

FIG. 5.3. *Storage for 256 Data Blocks*

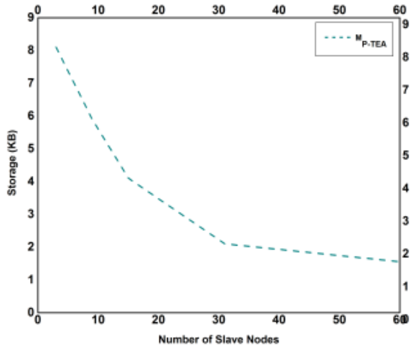

FIG. 5.4. *Storage versus Number of Slave Nodes*

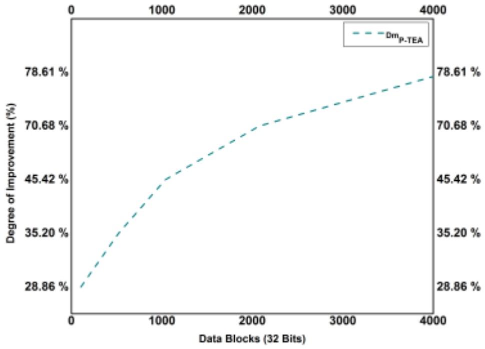

FIG. 5.5. *Degree of Improvement in Storage versus Data Blocks*

in parallel mode, the lesser the memory requirement.

Figure 5.4 shows the impact of the number of slaves on the storage in the proposed framework for 4160 blocks of 32-bit data. It is observed that the storage used is further reduced if the number of slave nodes for data processing is increased.

Figure 5.5 illustrates the degree of improvement in the storage of TEA implementation in a sequential approach versus the proposed data management framework. It is observed that the value of the degree of improvement in the proposed framework increases as the size of input data increases i.e. if there is huge data, the proposed framework proves to be more advantageous in saving resources in IoT networks. There is 28.86% to 78.61% improvement in the storage requirements of a node in the proposed framework as compared to the sequential approach (calculated by eq. 4.4).

Figure 5.6 shows the impact of cluster size on the processing time in the proposed data management framework for 4160 blocks of 32-bit data. It is observed that the processing time is reduced if the number of slave nodes for data processing is increased because the data size is reduced and time to execute on the same processor is divided by a factor N number of nodes.

Figure 5.7 shows the execution time of TEA on multiple nodes is less compared to the single node TEA implementation due to data parallelization for processing.

Figure 5.8 illustrates the degree of improvement in the processing time of TEA implementation in sequential versus the proposed approach. It is observed that the value of the degree of improvement in the proposed framework increases as the size of input data increases i.e. if there is huge data, the proposed framework proves to be more advantageous in saving the resources in IoT networks. There is 41.86% to 68.61% improvement in

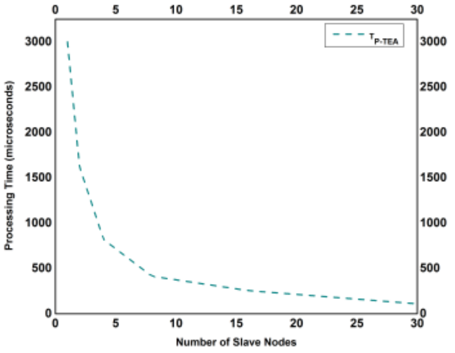

FIG. 5.6. *Processing Time versus Number of Slave nodes*

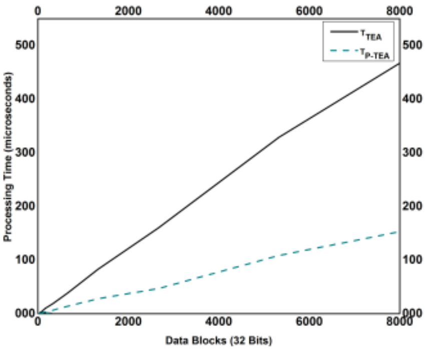

FIG. 5.7. *Processing Time versus Data Blocks*

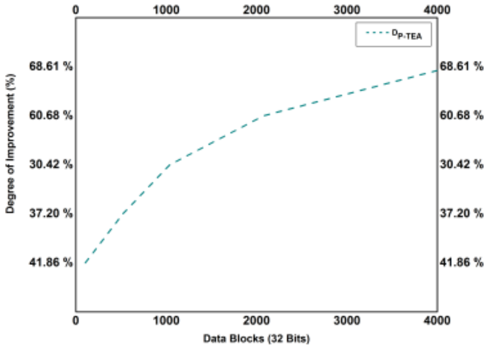

FIG. 5.8. *Degree of Improvement in Processing Time versus Data Blocks*

processing time in the proposed framework as compared to the sequential approach (calculated by eq. 4.5).

The proposed framework is evaluated using empirical data as well. Figure 5.9, 5.10 and 5.11 shows the energy versus time, storage versus the number of slave nodes, and processing time versus the number of slave nodes in the proposed framework as compared to the sequential approach.

It is evident from the Figures 5.9, 5.10 and 5.11 that the energy, storage and processing time requirements on the IoT device in the proposed framework are less as compared to a sequential approach.

6. Conclusions. In most of the pervasive applications of IoT, the resources of an IoT device are constrained and due to these resource limitations, there are restrictions on the use of data processing algorithms on an IoT node e.g. encryption and decryption of data causes an overhead of storage, processing, and energy to the existing data on an IoT node. To reduce these resource overheads in such applications, data management is important as data is the common factor that imposes restrictions on the use of resources. This paper proposed a data management framework that uses the concept of data parallelization on multiple nodes for minimizing the amount of data for processing on an individual IoT node and communication of data over longer distances.

From the comparative analysis of resources used in implementing a lightweight algorithm on the proposed framework and in sequential mode, the simulation results showed the proposed framework is better than the sequential one in terms of energy, storage, and processing time. Because of parallelization, the proposed framework results in fair utilization of resources and significant saving of resources. While in the case of a sequential approach, a particular node is overloaded while neighboring nodes are underutilized, resulting in fast depletion of resources especially energy which creates holes in the network. There is an improvement of 12.5% in

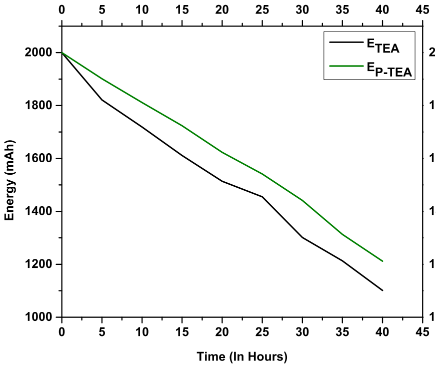

FIG. 5.9. *Energy versus Time*

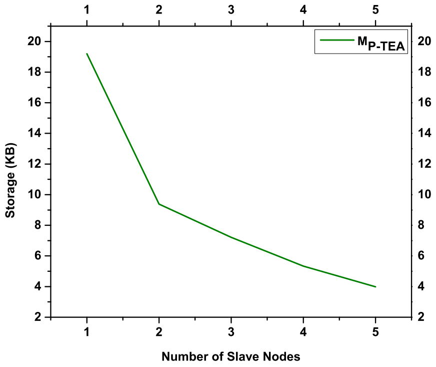

FIG. 5.10. *Storage versus Number of Slave nodes*

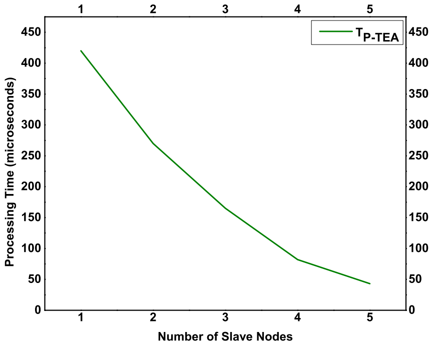

FIG. 5.11. *Processing Time versus Number of Slave nodes*

energy and 25% in the number of alive nodes in the proposed framework as compared to the sequential approach. We have observed that the storage and processing requirements of a node in the proposed framework is lesser and as the number of slave nodes for a master node increases to N, the storage and processing requirements of a node is decreased by a factor of N. There is 28.86% to 78.61% improvement in the storage of a node as the size of data on the master node increases to 4000 data blocks. Also, better performance has been observed in terms of processing time with 41.86% to 68.61% improvement in the network.

The proposed framework has been tested on the empirical data as well and it also showed improved resources in the proposed framework as compared sequential approach, making the proposed solution suitable to map with the real-life scenarios of pervasive IoT applications. We also conclude from the implementations that larger the number of slave nodes for the TEA implementation, lesser the resource consumption of a node for a particular interval of time but there has to be a limit for the division of work among the slave nodes in the cluster, otherwise it will increase the operational cost both at the node and network level.

In this paper, we have not compared the proposed data management framework with any other framework because the literature available for parallelization based data management frameworks for resource-constrained IoT applications is very scarce and the ones described in data parallelization approaches are mostly designed and implemented at hardware level only. One of the interesting future research developments of this paper lies in comparing the proposed parallelization based data management framework at the hardware level with the existing solutions. Also, the proposed framework can be extended to include more resource parameters and constraints to map with the real-world IoT scenarios.

REFERENCES

[1] Di Martino Et.al *Internet of things reference architectures, security and interoperability: A survey*, *Internet of Things* , 1, pp.99-112 (2018).

[2] S. Zahoor and R.N. Mir, *Resource management in pervasive Internet of Things: A survey*, *Journal of King Saud University-Computer and Information Sciences*, (2018).

[3] Z. Sheng Et.al, Lightweight management of resource-constrained sensor devices in internet of things, *IEEE internet of things journal*, 2(5), pp.402-411 (2015).

[4] S. Zahoor and R.N. Mir, *Resource management in pervasive Internet of Things: A survey*, *Journal of King Saud University-Computer and Information Sciences*, (2018).

[5] B. Guidi and L. Ricci, *Aggregation Techniques for the Internet of Things: An Overview,In The Internet of Things for Smart Urban Ecosystems*,pp. 151-176, Springer, Cham (2019).

[6] L. Jiang Et.al,*An IoT-oriented data storage framework in cloud computing platform*, *IEEE Transactions on Industrial Informatics*, 10(2), pp.1443-1451 (2014).

[7] P. Thubert Et.al, *6TiSCH centralized scheduling: When SDN meet IoT*, *In Proc. of IEEE Conf. on Standards for Communications & Networking (CSCN'15)*, (2015).

[8] A. Celesti Et.al, *Exploring container virtualization in IoT clouds*, *In 2016 IEEE International Conference on Smart Computing (SMARTCOMP)*, pp. 1-6, IEEE (2016).

[9] S. Li and L. Da Xu, *Securing the internet of things*, *Syngress*,(2017).

[10] J.T. Townsend, *Serial vs. parallel processing: Sometimes they look like Tweedledum and Tweedledee but they can (and should) be distinguished*, *Psychological Science*, 1(1), pp.46-54 (1990).

[11] M.J. Quinn,*Parallel programming*, TMH CSE, 526 (2003).

[12] B. Diene,*Data management techniques for Internet of Things*, *Mechanical Systems and Signal Processing*, 138, p.106564 (2020).

[13] H. Rahman Et.al, *Comparison of data aggregation techniques in Internet of Things (IoT)*, *In 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 1296-1300, IEEE (2016).

[14] B. Pourghebleh. and N.J. Navimipour,*Data aggregation mechanisms in the Internet of things: A systematic review of the literature and recommendations for future research*, *Journal of Network and Computer Applications*, 97, pp.23-34 (2017).

[15] T. Zhu Et.al, *An architecture for aggregating information from distributed data nodes for industrial internet of things*, *Computers & Electrical Engineering*, 58, pp.337-349 (2017).

[16] F. Xie,*CaCa: chinese remainder theorem based algorithm for data aggregation in internet of things on Ships*, *In Applied Mechanics and Materials*, 701, Trans Tech Publications Ltd, pp. 1098-1101, (2015).

[17] A. Alkhamisi Et.al,*A cross-layer framework for sensor data aggregation for IoT applications in smart cities*, *In 2016 IEEE International Smart Cities Conference (ISC2)*, IEEE, pp. 1-6 (2016).

[18] Z. Li, *Lifetime balanced data aggregation for the internet of things*, *Computers & Electrical Engineering*, 58, pp.244-264 (2017).

[19] A. Koike Et.al, *Iot network architecture using packet aggregation and disaggregation*, *In 2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, IEEE, pp. 1140-1145,(2016).

[20] Y. Liu Et.al *A novel trust-based secure data aggregation for internet of things*, *In 2014 9th International Conference on Computer Science & Education*, pp. 435-439, IEEE.

[21] S.G. Shilpa and S.M. Sundaram, *Data Aggregation Techniques Over Wireless Sensor Network-A Review*, (2013).

[22] Y. Qin Et.al, *When things matter: A survey on data-centric internet of things*, *Journal of Network and Computer Applications*, 64, pp.137-153 (2016).

[23] J. Ledlie Et.al, *Provenance-aware sensor data storage*, *In 21st International Conference on Data Engineering Workshops (ICDEW'05)*, pp. 1189-1189, IEEE (2005).

[24] W.J. Li Et.al, *Just IoT Internet of Things based on the Firebase real-time database*,*In 2018 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE)*, pp. 43-47, IEEE (2018).

[25] T. Fujita Et.al,*Centralized storage management method*, *U.S. Patent* 7,152,144 (2006).

[26] M.E. Rottsolk and S.P. Nolan, *Microsoft Corp*, *Centralized healthcare data management*, U.S. Patent Application 12/345,334 (2010).

[27] M. Kovatsch Et.al,*Moving application logic from the firmware to the cloud: Towards the thin server architecture for the internet of things*, *In Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, IEEE, pp. 751-756 (2012).

[28] R. Roman Et.al, *On the features and challenges of security and privacy in distributed Internet of Things*, *Computer Networks*, 57, 10, pp. 2266–2279 (2013).

[29] F. Zhu Et.al, *Private entity authentication for pervasive computing environments*, *International Journal of Network Security*, 14, 2, pp. 86–100 (2012).

[30] S. Shin Et.al, *An effective authentication mechanism for ubiquitous collaboration in heterogeneous computing environment*, *Peer-to-Peer Networking and Applications*, (2013).

[31] Y. Liu Et.al, *PKC based broadcast authentication using signature amortization for WSNs*, *IEEE Transactions on Wireless Communications*,11,6, pp. 2106–2115 (2012).

[32] M. Domingo-Prieto Et.al,*Distributed pid-based scheduling for 6tisch networks*, IEEE Communications Letters, 20(5), pp. 1006-1009 (2016).

[33] H. Gomaa, *Software modeling and design: UML, use cases, patterns, and software architectures*, *Cambridge University Press*, (2011).

[34] M. Kushwaha, *Oasis: A programming framework for service-oriented sensor network*, *In 2nd International Conference on Communication Systems Software and Middleware*, IEEE, pp. 1-8 (2007).

[35] T. G. Stavropoulos Et.al, *aWESoME: A web service middleware for ambient intelligence*, *Expert Systems with Applications*, 40(11), pp. 4380-4392 (2013).

[36] I. Corredor Et.al, *Knowledge-aware and service-oriented middleware for deploying pervasive services*, *Journal of Network and Computer Applications*, 35(2), pp. 562-576 (2012).

[37] M. S. Familiar Et.al, *Building service-oriented smart infrastructures over wireless ad hoc sensor networks: A middleware perspective*, *Computer Networks*, 56(4), pp. 1303-1328 (2012).

[38] L. Atzori, A. Iera and C. Giacomo Morabito, "The internet of things: a survey ", Comput. Netw., 2010.

[39] M.J. Scheepers, *Virtualization and containerization of application infrastructure: A comparison*, *In 21st Twente Student Conference on IT*, 21 (2014).

[40] R. Hussain Et.al, *Federated Edge Computing for Disaster Management in Remote Smart Oil Fields*, *In 2019 IEEE 21st International Conference on High Performance Computing and Communications, IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pp. 929-936, IEEE (2019).

[41] R. Petrolo Et.al, *The design of the gateway for the cloud of things*, *Annals of Telecommunications*, 72(1-2), pp.31-40, 2017.

[42] C. Pahl Et.al, *A container-based edge cloud paas architecture based on raspberry pi clusters*,*In 2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*, pp. 117-124, IEEE (2016).

[43] L.E. Li Et.al, *Toward software-defined cellular networks*, *In 2012 European Workshop on Software Defined Networking*, pp. 7-12, IEEE (2012).

[44] L.E. Li Et.al *Toward software-defined cellular networks*, *In 2012 European Workshop on Software Defined Networking*, pp. 7-12, IEEE (2012).

[45] C.Y. Chen Et.al, *Securing real-time internet-of-things*, *Sensors, 18(12)*, p.4356 (2018).

[46] M. Nagendra and M.C. Sekhar, *Performance improvement of Advanced Encryption Algorithm using parallel computation*, International Journal of Software Engineering and Its Applications, 8(2), pp.287-296 (2014).

[47] T. Balamurugan and T. Hemalatha, *Parallelization of Symmetric and Asymmetric Security Algorithms for Multi-Core Architectures*, *International Journal of Science and Research (IJSR)*, 3(12),(2014).

[48] W. Sun Et.al, *Security and privacy in the medical Internet of Things: A review*, *Security and Communication Networks*, (2018).

[49] A. Shah and M. Engineer, *A survey of lightweight cryptographic algorithms for iot-based applications*, *In Smart Innovations in Communication and Computational Sciences*, Springer, Singapore, pp. 283-293 (2019).

[50] J. Srinivas Et.al, *Secure and efficient user authentication scheme for multi-gateway wireless sensor networks*, *Ad Hoc Networks*, 54, pp.147-169 (2017).

[51] F. Wu Et.al, *A lightweight and privacy-preserving mutual authentication scheme for wearable devices assisted by cloud server*, *Computers & Electrical Engineering*, 63, pp.168-181, 2017.

[52] Y. Qiu and M. Ma, *A mutual authentication and key establishment scheme for m2m communication in 6lowpan networks*, *IEEE transactions on industrial informatics*, 12(6), pp.2074-2085 (2016).

[53] R. Giuliano Et.al, *Security access protocols in IoT capillary networks*, *IEEE Internet of Things Journal*, 4(3), pp.645-657 (2016).

[54] I. Butun Et.al, *A survey of intrusion detection systems in wireless sensor networks*, *IEEE communications surveys & tutorials*, 16(1), pp.266-282 (2013).

[55] W. Chen, *An ibe-based security scheme on internet of things*, *in: Cloud Computing and Intelligent Systems (CCIS), 2012 IEEE 2nd International Conference*, 3, IEEE, pp. 1046–1049 (2012).

[56] C. Perera Et.al, *Context aware computing for the Internet of Things: a survey*, *IEEE Commun. Surv. Tutor.*, 16 (1), pp. 414–454(2014).

[57] T. Choi Et.al, *The best keying protocol for sensor networks*, *Pervasive and Mobile Computing*, 9(4), pp.564-571 (2013).

[58] Z. Wang Et.al, *Secure and efficient control transfer for IoT devices*, *International Journal of Distributed Sensor Networks*, 9(11), pp.503404 (2013).

[59] M.R. Abdmeziem and D. Tandjaoui, *An end-to-end secure key management protocol for e-health applications*, *Comput. Electr. Eng.*, 44, 184–197 (2015).

[60] F. Al-turjman and M. Gunay, *CAR Approach for the Internet of Things, approche de la CAR pour l internet des objects*, *Can. J. Electr. Comput. Eng.*, 39 (1), 11–18 (2016).

[61] R. Neisse Et.al, *SecKit: a Model-based Security Toolkit for the Internet of Things*, *Comput. Secur.*, 54, 60–76 (2015).

[62] X. Yao Et.al, *A lightweight attribute-based encryption scheme for the Internet of Things*, *Future Gener. Comput. Syst.*, 49, 104–112 (2014).

[63] N. Oualha and K.T. Nguyen, *Lightweight attribute-based encryption for the internet of things*, *in: 2016 25th International Conference on Computer Communication and Networks (ICCCN)*, IEEE, pp. 1–6 (2016).

[64] L. Touati Et.al, *C-cp-abe: Cooperative ciphertext policy attribute-based encryption for the internet of things*, *In 2014 International Conference on Advanced Networking Distributed Systems and Applications*, pp. 64-69, IEEE (2014).

[65] F. Restuccia Et.al, *Securing the internet of things: New perspectives and research challenges*, (2018).

[66] L. Da Xu Et.al, *Internet of things in industries: A survey*, *IEEE Transactions on industrial informatics*, 10(4), pp.2233-2243 (2014).

[67] C. Kan Et.al, *Parallel computing and network analytics for fast Industrial Internet-of-Things (IIoT) machine information processing and condition monitoring*, *Journal of manufacturing systems*, 46, pp.282-293 (2018).

[68] P.P. Ray *A survey on Internet of Things architectures*, *Journal of King Saud University-Computer and Information Sciences*,

30(3), pp.291-319 (2018).

[69] N.S. Raghavan and T. Waghmare, *DPAC: an object-oriented distributed and parallel computing framework for manufacturing applications*, *IEEE transactions on robotics and automation*, 18(4), pp.431-443 (2002).

[70] D. Mourtzis Et.al, *A cloud-based approach for maintenance of machine tools and equipment based on shop-floor monitoring*, *Procedia Cirp*, 41, pp.655-660 (2016).

[71] F. Tao eT.AL, *FC-PACO-RM: a parallel method for service composition optimal-selection in cloud manufacturing system*, *IEEE Transactions on Industrial Informatics*, 9(4), pp.2023-2033 (2012).

[72] V. Venugopal and D. Manikantan Shila, *Hardware acceleration of TEA and XTEA algorithms on FPGA, GPU and multi-core processors*, *In Proceedings of the ACM/SIGDA international symposium on Field programmable gate arrays*, ACM, pp. 270-270 (2013).

[73] P. Yalla and J.P. Kaps, *Lightweight cryptography for FPGAs*, *In 2009 International Conference on Reconfigurable Computing and FPGAs*, pp. 225-230, IEEE (2009).

[74] S. Kerckhof Et.al, *Towards green cryptography: a comparison of lightweight ciphers from the energy viewpoint*, *In International Workshop on Cryptographic Hardware and Embedded Systems*, Springer, Berlin, Heidelberg, pp. 390-407 (2012).

[75] S. Kim and S. Kim, *User preference for an IoT healthcare application for lifestyle disease management*,*Telecommunications Policy*, 42(4), pp.304-314 (2018).

[76] J. Sherly and D. Somasundareswari, *Internet of things based smart transportation systems*, *International Research Journal of Engineering and Technology*, 2(7), pp.1207-1210 (2015).

[77] S. Liniger and B. Stiller, *Parking prediction techniques in an iot environment*, Master's thesis (2015).

[78] F. Montori Et.al, *A collaborative internet of things architecture for smart cities and environmental monitoring*, *IEEE Internet of Things Journal*, 5(2), pp.592-605 (2017).

[79] V. Vishwarupe Et.al, *Zone specific weather monitoring system using crowdsourcing and telecom infrastructure*, *In 2015 International Conference on Information Processing (ICIP)*, pp. 823-827, IEEE (2015).

[80] J.M Talavera Et.al, *Review of IoT applications in agro-industrial and environmental fields*, *Computers and Electronics in Agriculture*, 142, pp.283-297 (2017).

[81] T. Qiu Et.al, *Framework and case studies of intelligence monitoring platform in facility agriculture ecosystem*, *In 2013 Second International Conference on Agro-Geoinformatics (Agro-Geoinformatics)*, pp. 522-525, IEEE (2013).

[82] S. Nagarkar, *IOT concept, technologies and applications for smart homes-A Review*, *Conference proceedings CTIcon* (2017).

Edited by: Anand Nayyar
Received: Apr 27, 2020
Accepted: Jul 17, 2020