International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 10 | Oct 2023 www.irjet.net p-ISSN: 2395-0072

Green Computing for Internet of Things: Energy Efficient and Delay-Guaranteed Allocation

UDAY M. KHARATMOL[1], PRADNYA N. SHINDE[2]

[1]UG student, Dept. of Computer Engineering, $I^{2}IT$ College, Maharashtra, India
[2] Professor, Dept. of Computer Engineering, $I^{2}IT$ college, Maharashtra, India

---***---

Abstract - *The increasing demand for IoT is driving the development of green cloud computing, focusing on energy-efficient practices and delay-guaranteed workload allocation strategies. This approach aims to balance computational demands with sustainability goals, considering the ecological impact of cloud-based services and data centers. The study explores the importance of green computing in a sustainable IoT ecosystem, proposing a systematic framework for energy consumption minimization in an IoT-edge-cloud computing system. It also analyzes the Lyapunov drift-plus penalty properties of edge server queuing systems, proposing a delay-based scheme to minimize drift-plus-penalty.*

Key Words: Internet of Things (IoT), cloud computing, edge computing, energy consumption, Model Description, DBWA Algorithm, Application of Green Cloud Computing

1.INTRODUCTION

Internet of Things (IoT) brings together smart objects integrated into a heterogeneous network for monitoring and decision-making process. In the ever-expanding digital landscape of the 21st century [1], cloud computing has emerged as a cornerstone technology, revolutionizing the way individuals and organizations access and harness computational resources.

Cloud services offer unparalleled scalability, flexibility, and cost-effectiveness, enabling diverse applications ranging from data storage to machine learning. However, this rapid proliferation of cloud services has given rise to a critical concern: their substantial environmental footprint. Green cloud computing represents a transformative approach to address this challenge, offering a sustainable path forward for cloud technology. At its core, green cloud computing aims to reconcile the soaring computational demands of modern society with environmental stewardship.

This paradigm encompasses two vital components: energy-efficient resource management and delay-guaranteed workload allocation.

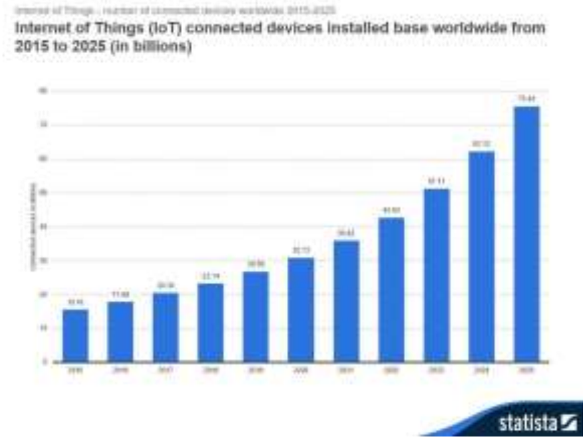

Figure 1: IoT Connected Devices Statistic

2. BACKGROUND

2.1 Green computing

Green cloud computing is a technology approach that aims to reduce the environmental impact and energy consumption of cloud services and data centers while maintaining their efficiency. It focuses on strategies like energy efficiency, renewable energy sources, workload optimization, and reducing carbon emissions. Green cloud computing is crucial for sustainable and environmentally responsible computing practices.

Green cloud computing focuses on optimizing the use of energy in data centers. This includes strategies like server virtualization, dynamic resource allocation, and advanced cooling techniques to reduce power consumption. Many green cloud initiatives incorporate renewable energy sources, such as solar and wind power, to reduce reliance on fossil fuels and minimize carbon emissions. Green cloud computing strives to allocate computational resources efficiently, ensuring that servers are fully utilized while minimizing energy waste. This involves techniques like workload consolidation and load balancing. Green cloud computing addresses Quality of Service (QoS) requirements, particularly for latency-sensitive applications. It aims to provide timely and reliable service delivery, often through real-time workload allocation.
Sustainable data center design plays a crucial role in green cloud computing. Companies may invest in eco-friendly data

center construction, including efficient cooling systems, energy-efficient hardware, and water usage optimization. Green cloud computing is essential not only for reducing the environmental impact of technology but also for cost-effectiveness and long-term sustainability. By adopting green practices, cloud providers and users can contribute to a more eco-friendly and responsible computing ecosystem [2].

Figure 2: Green IoT Applications

2.2 Internet of things

IoT is an interconnection of devices building a smart world. It is a paradigm that impacts both society and technology. The Internet of Things (IoT) is a transformative technological paradigm that has reshaped the way we interact with the digital world and the physical environment around us. At its core, IoT is a network of interconnected devices, sensors, and objects that can communicate with each other and exchange data over the internet without requiring human intervention. These "smart" devices can range from everyday household items like thermostats and refrigerators to industrial machinery, vehicles, and wearable gadgets. Green computing has to focus on the reduction of energy consumption to meet the sustainability of the smart city and also be eco-friendly [3].

IoT has become a significant component in today's world that interconnects devices that are important in making decisions. The majority of these devices are sensors and equipment that facilitate the exchange of data through various networks enabling the device to device information sharing. This has resulted in the emergence of edge computing to achieve low latency response and alleviate resource congestion on the centralized data centers. It transfers all the device data computation to an edge data center near to them. The distributed infrastructure also balances network congestion created during transmission of data [3].

Figure 3: IoT Applications

The Internet of Things (IoT) is a technology paradigm that connects everyday objects, devices, and sensors to the internet, enabling them to collect, transmit, and exchange data. IoT has the potential to revolutionize various industries by providing real-time insights, automation, and remote control over physical objects. IoT architecture typically consists of several architecture layers [4], each serving a specific purpose in the flow of data and communication. This layer comprises sensors, actuators, and devices that interact with the physical world. Devices include a wide range of IoT endpoints, from simple sensors in household appliances to complex machinery in industrial settings.

The network layer handles the communication between IoT devices and bridges the gap between the perception layer and the subsequent layers. Various communication protocols, such as Wi-Fi, Bluetooth, Zigbee, LoRa, and cellular networks, are used to connect devices to the internet or local networks. Gateways may also be employed to aggregate data from multiple devices and establish connections to the internet or cloud services. The application layer encompasses the software and services that make use of IoT data to provide value to users or organizations. IoT applications can serve various purposes, such as home automation, industrial monitoring and control, healthcare, agriculture, and smart cities.

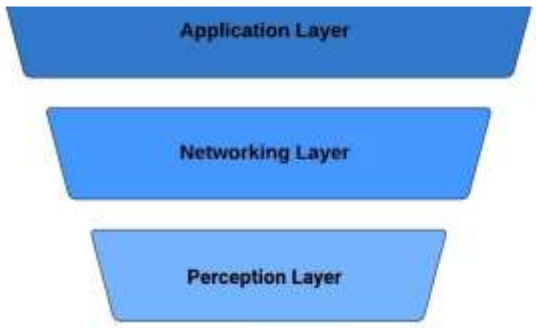

Figure 4: IoT classification based on power consumption.

2.3 Cloud computing

Cloud computing is a transformative technology that has revolutionized the way organizations and individuals' access, store, and manage digital resources and services. It enables users to access and use computing resources, such as servers, storage, databases, networking, software, and more, over the internet, often referred to as "the cloud." This cloud-based approach offers numerous advantages over traditional on-premises computing, including scalability, cost-efficiency, flexibility, and accessibility. The life cycle of Green IoT (Internet of Things) encompasses various stages, each with a focus on sustainability, energy efficiency, and environmentally responsible practices. An overview of the key stages with an emphasis on green design, green production, green utilization, and green recycling are shown in fig 5.

Cloud computing has resulted in the emission of CO_2 due to the energy consumption from the data centers. Various practices have been adopted to lower the energy consumption by data center machines by using hardware virtualization and energy-Conversant strap in software applications. The energy consumption is predicted to rise with the continuous usage of cloud computing services and the data centers which host them. It is for this energy concern that there is a need to rethink how data centers adopt green computing, and the equipment been used [6]. An overview of IoT data been collected from devices,processed, and analyzed, is shown in Figure 6.

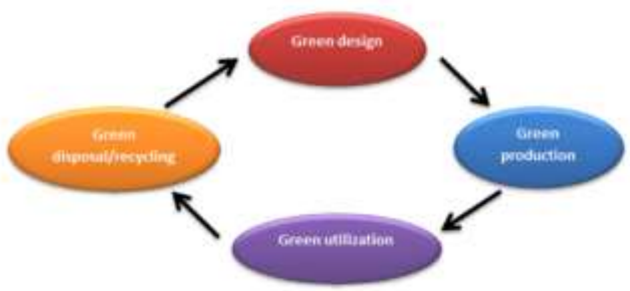

Figure 5: Life Cycle of Green IoT

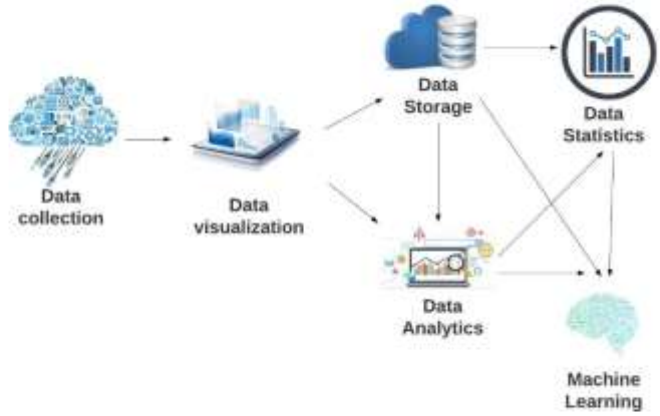

Figure 6: Overview of IoT

2.4Edge-IoT

An increase in the rise of mobile devices has resulted in mobileedge computing (MEC) for low latency responses. MEC provides mobile computing, network congestion control, and storage capacity to the edges of the networks. MEC lowers the usage of mobile energy and supports latency-critical applications. The development of the 5G network has been motivated by the gains of MEC, which combines both wireless communications and mobile computing to offload network computation. Wireless sensor networks are responsible for sending data by indoor devices, at the front end of Wireless Mesh Sensor Networks (WMNs), edge devices are deployed to reduce the network congestion helping users to tailor their needs through MEC [6].

3. Model Description

This section briefly introduces the system, traffic, delay and energy consumption models as well as problem formulation.

3.1 System Mode

As illustrated in Fig. 7, this paper considers an IoT-edge-cloud computing system with IoT regions, edge nodes and a cloud. Each of IoT regions endows with an edge node and a limited number of IoT devices. The edge node is the integration of an edge server and edge communication infrastructures. The edge server and the cloud have distinct computing capabilities. The IoT devices from IoT regions generate computation jobs stochastically. All the computation jobs from an IoT region are delivered to the edge node deployed in this region. The edge node makes workload allocation decisions for arrival jobs on computing locally, offloading to a neighbor edge or offloading to the cloud for computing. Notice that, to avoid ping-pong effect, we assume that when a job offloads from an edge node to a neighbor edge, it cannot be offloaded again.

We consider that there are M number of IoT regions and M number of edge nodes, each of which configures with an independent computing capability $P_i^{F\,1}$ for i ∈ M, where M= {0, 1, · · ·, M −1}is the IoT region space as well as edge node space.

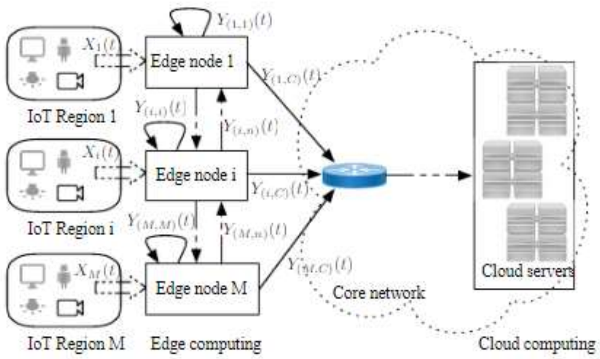

Figure 7: Overall architecture of an IoT-edge-cloud system

3.2 Traffic Model

A dynamic workload model is considered: (1) the computation jobs are generated from each of IoT regions stochastically and independently; (2) in each region, the number of computation jobs per time slot follows an independent and identical distribution (i.i.d), and the sizes of jobs belonging to the same type also follow an i.i.d. Notice that, the size of a job is measured in bits in this paper. Then, the CPU cycles required to execute a job with size S can be derived by

$$Cw = \chi S, \quad (1)$$

The term computing capability refers to the maximum rate at which the server can process a computation task, e.g, $P_i^F \sim f_i^F$, where f_i^F is the CPU-cycle frequency. We assume that $f_i^F \leq f_i^{max}$. where χ is the number of CPU cycles required to process one bit of a job.

As shown in Fig. 7, let Xi(t) be the number of computation jobs generated from the i^th IoT region that arrive at edge node i in slot t[2] and $Xwi(t) = \sum_{k=0}^{Xi(t)-1} S^k_i(t)$ be the corresponding workload, where $S^k_i(t)$ is the size of the k^th job. Let $\lambda_i = E[Xi(t)]$ be the long term job generation rate and $Si = E[Xwi(t)/Xi(t)]$ be the expected size of jobs generated in the i^th region. Let $Y_{(i,i)}(t)$, $Y_{(i,n)}(t)$ and $Y_{(i,C)}(t)$ be the number of jobs belonging to the job set of {0, 1, · · · , Xi(t) − 1} that are determined to compute in the local edge node i, neighbor edge node n ∈ M and the cloud, respectively.

Let $Yw_{(i,i)}(t)= \sum_{k=0}^{Y(i,i)(t)-1} S^k_i(t)$, $Yw_{(i,n)}(t)=\sum_{k=0}^{Y(i,n)(t)-1} S^k_i(t)$ and $Yw_{(i,C)}(t)= \sum_{k=0}^{Y(i,C)(t)-1} S^k_i(t)$ be the corresponding workload.

Then, for i ∈ M, we have

$$\begin{cases} Xi(t) = Y(i,i)(t) + Y(i,n)(t) + Y(i,C)(t), \\ Xwi(t) = Yw(i,i)(t) + Yw(i,n)(t) + Yw(i,C)(t). \end{cases} \quad (2)$$

3.2 Delay Model

Job may experience two types of delays, including transmission delay and computation delay.

3.2.1 Transmission Delay

We consider two types of network transmission paths, including the path from an edge node to its neighbor edge node and the path from an edge node to the cloud. Let $bw_{(i,j)}$ denote the bandwidth of the transmission path from the node i (e.g., an edge node) to the node j (e.g., a neighbor edge node, or, the cloud). Let S^k be the size of the job k that transmits over the path. Then, the transmission delay of the job in the path is derived by

$$D^k_{\text{comm(i,j)}} = \alpha(i,j) + \frac{S^k}{bw(i,j)}, \quad (3)$$

where α(i,j) is the factor of communication delay, e.g., the network delay introduced by network congestion.

3.2.2 Computation Delay

Due to computation resource constraints in edge nodes, we assume a queuing subsystem for each of the edge servers. Let Qi(t) be the number of jobs queuing in the subsystem i at the beginning of slot t. Then, the evolution of the queue length Qi follows

$$Q_i(t+1) = max[Q_i(t) + Y_i(t) - r_i(t), 0], \quad (4)$$

where $Y_i(t)$ and $r_i(t)$ are the number of jobs arrive and service respectively at the subsystem i during slot t. $Y_i(t)$ is derived by

$$Y_i(t) = \sum_{j\in M} Y_{(j,i)}(t). \quad (5)$$

Let $Qw_i(t)$ be the corresponding workload considering the number of jobs as well as job sizes queuing in the subsystem Let $Qw_i(t)$ be the corresponding workload considering the number of jobs as well as job sizes queuing in the subsystem

$$Qw_i(t+1) = \max[Qwi(t) + Ywi(t) - \frac{P_i^F}{\chi}, 0], \quad (6)$$

Where $\frac{P_i^F}{\chi}$ is the bits that the server can process in a slot; $Yw_i(t)$ is the aggregated workload arrives in slot t, which is derived by

$$Yw_i(t) = \sum_{k=0}^{Y_i(t)-1} S_i^k, \quad (7)$$

where S_i^k is the size of the k^th job that arrives in slot t. Let K be the job space that consists of Yi(t) number of jobs that arrive in slot t for t = 0, 1, · · ·, ∞ in subsystem i (e.g., an edge node, or, the cloud).

Then, the computation delay of the k^th (k ∈ K) job with size S_i^k that arrives at the queuing subsystem i ∈ M in slot t could be derived by

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 10 | Oct 2023 www.irjet.net p-ISSN: 2395-0072

$$D_{\text{comp}}^{(\text{i,k})}(t) = D_{\text{que}}^{(\text{i,k})} + \frac{\chi S_i^k}{P_i^F}, \qquad (8)$$

here $D_{\text{que}}^{(\text{i,k})}$ is the queuing delay, which could be approximated by

$$D_{\text{que}}^{(\text{i,k})}(t) = \frac{\chi(\text{Qw}_i(t) + \sum_{j=0}^{k-1} S_i^j}{P_i^F}, \qquad (9)$$

where $\sum_{j=0}^{k-1} S_i^j$ is the workload that arrives in slot t and ahead of the k^{th} job. As for a job offloading to the cloud, we assume that the job can be computed immediately after its arrival at the cloud. Thus, the computation delay of the k^{th} job with size S^k that arrives at the cloud in slot t is derived by

$$D_{\text{comp}}^{(\text{C,k})}(t) = \frac{\chi S^k}{P^c}. \qquad (10)$$

3.2.3 End-To-End Delay

Accordingly, for the k^{th} job with size S_i^k that arrives at the computing node i (e.g., the local edge node, the neighbor edge node, or, the cloud) in slot t and determines to compute at this node, assuming the job was generated from IoT region j, the e2e delay of the job cloud be derived by

$$D^{(i,k)}(t) = I_F^k(t) D_{\text{comp}}^{(\text{i,k})}(t) + I_{NF}^k(t)[D_{\text{comm(j,i)}}^k + D_{\text{comp}}^{(\text{i,k})}(t)]$$
$$+ I_C^k(t)[D_{\text{comm(j,C)}}^k + D_{\text{comp}}^{(\text{C,k})}(t)], \quad (11)$$

where $D_{\text{comp}}^{(\text{i,k})}(t)$ and $D_{\text{comp}}^{(\text{C,k})}(t)]$, are the computation delays in the edge node I and the cloud C, respectively; $D_{\text{comm(j,i)}}^k$ and $D_{\text{comm(j,C)}}^k$ are the transmission delays in paths of local edge node-neighbor edge node and local edge node-the cloud that the job may experience, respectively; $I_F^k(t), I_{NF}^k$ and $I_C^k(t)$ are mutually exclusive binary computation decision indicators for the k th job in slot t. When the job is determined to compute in the local edge node, $I_F^k(t)$ = 1; when it is determined to compute in the neighbor edge node, I_{NF}^k = 1; it is determined to compute in the cloud, $I_C^k(t)$ = 1; otherwise, the indicators are set to zeros. Thus, the computation decision indicators should satisfy the following constraint:

$$I_F^k(t) + I_{NF}^k + I_C^k(t) = 1, \quad \forall k \in K, t = 0, \ldots, \infty. \quad (12)$$

4. DBWA

According to the Lyapunov drift theory, if an algorithm can be designed to control the Lyaponov drift-plus-penalty ΔL(t)+ VE[Pw(γ (t))|Qw(t)] as described in inequality in below towards negative,

ΔL(t)+ VE[Pw(γ (t))|Qw(t)] ≤

$$B + \text{VE}[\text{Pw}(\gamma\,(t))|\text{Qw}(t)] - \sum_{i \in M} \text{Qw}_i(t) \frac{P_i^F}{X} +$$
$$\sum_{j \in M} \text{Qw}_i(t) \text{E}[\text{Yw}_i(t)|\text{Qw}(t)](t) \qquad (13)$$

then the queue Qw(t) would be stable while the optimal E[Pw] would be approximated. Therefore, based on the results of Lemmas 1-2, we propose a delay-based workload allocation (DBWA) algorithm to find out a sequential optimal workload allocation decisions γ ∗ (t) for t = 0, · · ·, ∞ to minimize a bound on the right-hand side of inequality. (13) every slot, such that minimize a bound on ΔL(t)+VE[Pw(γ (t))|Qw(t)]. The detail of DBWA is shown in Algorithm 1.

In DBWA, since for the jobs generated from IoT region i, the offloading decisions are local edge i, neighbor edge n, or the cloud, we have $Yw_{(i,j')}(t) = 0$ for j′ ∈ M − {i, n}. Accordingly, we use $\sum_{j \in \{i,n\}} Qw_j(t) Yw_j(t)$ in Eq. (14) instead of P j∈M $Qw_j(t) Yw_j(t)$, as shown in Algorithm 1. This paper assumes that the edge nodes can obtain the workload states (e.g., Q and Qw) of other nodes at most once in a slot. Thus, an edge node cannot obtain the updated job arrival events of other nodes within a slot, e.g., the edge node i cannot obtain $Yw_{(j,n)}(t)$ for j ∈ M − {i} in slot t. Accordingly, we use Eq. (16) in Algorithm 1 to approximate the workload of the neighbor node.

Generally, a job in an IoT-edge-cloud system experiences two processes, including workload allocation and scheduling. Workload allocation process determines where to computation offload the job, while scheduling services the job based on the computation offloading decision. The detail of our proposal for handling jobs in the IoT-edge-cloud system based on DBWA is described in Algorithm 2.

Algorithm 1 Delay-Based Workload Allocation Algorithm (DBWA)

Input: Xi(t), Qw(t).

1) Initialization: $Yw_{(i,i)}(t) = 0$, $Yw_{(i,n)}(t) = 0$ for n ∈ {i's neighbor list}.
2) Decision process: Choose γ ∗ i (t) as the solution to the following:
Minimize:

$$VPw(\gamma_i(t)) + \sum_{j \in \{i,n\}} \text{Qw}_j(t) Yw_j(\gamma_i(t)) \qquad (14)$$

Subject to:

(2),

$$\begin{cases} Y_{(i,i)}(t) = \sum_{k \in K_i^X} I_F^{(i,k)}(t), \\ Y_{(i,n)}(t) = \sum_{k \in K_i^X} I_{NF}^{(i,k)}(t), \\ Y_{(i,C)}(t) = \sum_{k \in K_i^X} I_C^{(i,k)}(t). \end{cases} \qquad (15)$$

$$\begin{cases} Yw_{(i,i)}(t) = \sum_{k\in K_i^X} I_F^{(i,k)}(t)S_i^k, \\ Yw_{(i,n)}(t) = \sum_{k\in K_i^X} I_{NF}^{(i,k)}(t)S_i^k, \\ Yw_{(i,C)}(t) = \sum_{k\in K_i^X} I_C^{(i,k)}(t)S_i^k. \end{cases} \quad (16)$$

$$Yw_i(t) = \sum_{j\in M, j\neq i} Yw_{(j,i)}(t) + Yw_{(i,i)}(t), \quad (17)$$

$$Yw_n(t) = Yw_{(i,n)}(t), \quad (18)$$

$$D^{(i,k)}(t) = D^{(i,k)}. \quad (19)$$

where Eq. (2) is the traffic constraint; Eqs. (15)-(16), (17)-(18) follow the definition of γ (t).

Default: if no solution satisfies Eq. (19), set $I_F^{(i,k)}(t) = I_{NF}^{(i,k)}(t) = 0$ and $I_C^{(i,k)}(t) = 1$.

3) Processing the decision: observe γ_i^* (t), for $k \in K_i^X$,**do**

a) If $I_F^{(i,k)}(t) = 1$: buffer the job into the local queuing system;

b) Else if $I_{NF}^{(i,k)}(t) = 1$: transmit the job to the neighbor edge node n;

c) Else: transmit the job to the cloud.

Output: γ_i^* (t).

Algorithm 2 DBWA-Based Workload Allocation and Scheduling Processes

Initialization: Q(0) = 0, Qw(0) = 0. Every slot t ≥ 0, **do**

1) **Workload allocation process:**

a) The edge node i (i ∈ M) receives $X_i(t)$ number of jobs that were generated from IoT region i, **do**

i) Observe Qw(t), and initiate Algorithm 1 for edge node i;

ii) Calculate $Yw_{(i,i)}(t)$ and $Yw_{(i,i)}(t)$with Eqs.(15)-(16), and update $Q_i(t)$ and $Qw_i(t)$ with Eq. (20).

$$\begin{cases} Q_i(t+1) = Q_i(t) + Y_{(i,i)}(t), \\ Qw_i(t+1) = Qw_i(t) + Yw_{(i,i)}(t). \end{cases}$$
(20)

b) The edge node i (i ∈ M) receives $Y_{(j,i)}(t)$ (j ∈ M and j≠i)number of jobs with workload$Yw_{(j,i)}(t)$ that were offloaded from edge node j, **do**

i) Buffer the jobs into the local queuing system;

ii) Update $Q_i(t)$ and $Qw_i(t)$ with Eq. (21).

$$\begin{cases} Q_i(t+1) = Q_i(t) + Y_{(j,i)}(t), \\ Qw_i(t+1) = Qw_i(t) + Yw_{(j,i)}(t). \end{cases} \quad (21)$$

2) **Scheduling process:**

a) Scheduling in edge node i(i ∈ M):

i) Process the waiting jobs with service rate P_i^F in first-in-first-out (FIFO) discipline;

ii) Update $Q_i(t)$ and $Qw_i(t)$ with Eq. (22).

$$\begin{cases} Q_i(t+1) = max[Q_i(t) - r_i(t), 0], \\ Qw_i(t+1) = \max[Qwi(t) - \frac{P_i^F}{x}, 0]. \end{cases} \quad (22)$$

where $r_i(t)$ represents the number of jobs that are processed in edge node i in slot t.

b) Scheduling in the cloud:

i) Cloud receives $\sum_{i\in m}\sum_{k=0}^{Y_{i,C)}((t)-1} number$ of jobs at the beginning of slot t;

ii) Initiate P i∈M PY(i,C)(t)−1 k=0 number of VMs to process these jobs with service rate P C, respectively.

5. APPLICATION OF GREEN COMPUTING IN IOT

Green computing refers to the practice of designing, using, and disposing of computer systems and electronic devices in an environmentally friendly and energy-efficient manner. When applied to the Internet of Things (IoT), green computing principles can have a significant positive impact on both the environment and the overall sustainability of IoT deployments. Here are several ways in which green computing can be applied in IoT:

5.1 Autonomous vehicles

The automobile industry is investing in autonomous vehicles that analyze and make decisions based on their surroundings' data. These vehicles need to transmit data to manufacturers for usage tracking and maintenance alerts. However, this data transmission can cause congestion. To achieve low latency, manufacturers are implementing edge computing to reduce latency and energy usage for sensors, promoting an eco-friendly approach and reducing carbon emissions. Edge data centers are being used to facilitate this process.

5.2 Smart cities

City leadership utilizes massive data from sensors like traffic, infrastructure, and home appliances to address challenges. This data requires extensive computing capabilities for processing and analysis. Real-time responses to these devices reduce energy usage, ensuring efficient city management.

5.3Industries

IoT edge computing can be utilized in industries like oil drilling to gather data on environmental factors, reducing energy consumption in production by eliminating the need for pre-collected historical data.

6. CHALLENGES FACING GREEN COMPUTING IMPLEMENTATION

6.1 Green computing awareness

Studies reveal that only 28% of people are aware of the environmental impact of CO_2 emissions due to their lack of knowledge about green computing.

6.2 Equipment cost

Companies are considering adopting green computing, despite incurring fees, as traditional methods may not be as energy-efficient as modern methods, as they aim to control emissions and reduce energy consumption and CO_2 footprint.

Performance degradation

Concerns about eco-friendly equipment materials causing performance degradation necessitate education on their usage and performance.

7. SOLUTIONS TO THE CHALLENGES

7.1 People awareness

Many people are unaware of green computing principles, and it's crucial to educate them about energy consumption and recycling methods to promote environmental conservation.

7.2 Data centers

Data centers are crucial in the cloud computing industry, and their energy consumption should be regularly reviewed and biodegradable hardware components used.

7.3 Virtualization

Virtualization creates multiple simulated environments on a physical host, abstracting hardware, enabling efficient resource management, monitoring, resource sharing, and low energy utilization.

7.4 Recycling equipment

Discarding unwanted hardware should be biodegradable for eco-friendliness, as many computer parts can harm the environment. Using recyclable materials can reduce the impact of these materials.

8. CONCLUSION

This paper has studied Green Computing is essential for a sustainable IoT ecosystem, as it contributes to recyclable devices and reduces energy consumption. This paper investigates the energy-efficient and delay-guaranteed workload allocation problem in an IoT-edge-cloud computing system. A systematic framework is developed, including system, traffic, delay, and energy consumption models, to address the issue of energy consumption minimization. A delay-based workload allocation scheme, DBWA scheme, is proposed to minimize drift-plus-penalty and achieve the goal of minimizing energy consumption while providing per-job granular delay guarantee. The DBWA scheme is formulated to address the energy consumption problem in IoT-edge-cloud computing systems.

REFERENCES

[1] Harshal Gajanan Patil, Rasika Vishnu Tapase, Asst.prof.P.S.Gade, Asst.prof.V.V.Kadam."Green Computing for Internet of Things." IRJET, 2022.

[2] Arshad, Rushan, et al. "Green IoT: An investigation on energy saving practices for 2020 and beyond." IEEE Access 5 (2017): 15667-15681.

[3] Albreem, Mahmoud AM, et al. "Green internet of things (IoT): An overview." *2017 IEEE 4th ICSIMA*.

[4] Guo, Mian, Lei Li, and Quan sheng Guan. "Energy-Efficient and Delay Guaranteed Workload Allocation in IoT-Edge-Cloud Computing Systems." IEEE Access 7 (2019): 78685-78697.

[5] Albreem, Mahmoud AM, et al. "Green internet of things (IoT): An overview." 2017 IEEE 4th ICSIMA.

BIOGRAPHIES

UDAY M. KHARATMOL,
UG Student
International Institute of Information Technology, Pune.
SPPU university.

PRADNYA N. SHINDE
Assistant Professor
International Institute of Information Technology, Pune.
SPPU university.