Technical Journal, University of Engineering and Technology (UET) Taxila, Pakistan Vol. 28 No. 4-2023
ISSN:1813-1786 (Print) 2313-7770 (Online)

MCDM-Based Optimal Gateway Selection in Mobile Fog Computing

K. I. Malik[1], H. Arooj[2], B. Jamil[3], K. Munir[4], H. Ijaz[5]

[1] *Department of Computing and Innovation, Riphah International University, Lahore Campus, Pakistan*
[2] *Department of Mathematics and Statistics , University of Lahore, Sargodha, Pakistan*
[3,5] *Department of Computer Science and Information Technology, University of Sargodha, Sargodha, Pakistan*
[4] *Department of Computer Science, NUCES, Islamabad, Pakistan*

[3]*bushra.jamil@uos.edu.pk*

Abstract- The use of ubiquitous and pervasive computing and technological advancements have made mobile Internet of Things, an essential component across many next-generation domains, like land vehicular networks, transportation systems, and maritime networks. These mobile IoT applications demands, which call for persistent connectivity between mobile IoT devices and IoT infrastructure, have led to the evolution of mobile fog computing. Mobile fog computing makes fog services available everywhere within the coverage of fog networks, even when users and mobile IoT devices are mobile. The massive amounts of data produced by mobile IoT devices must first be preprocessed by choosing suitable mobile fog nodes that serve as gateways before being transferred to other fog nodes and the cloud. However, the dynamic context factors, high workloads, heterogeneity, reliability, and mobility of the participating objects make gateway selection in mobile fog computing a significant issue. Therefore, in this article, we propose a novel multicriteria decision-making (MCDM) algorithm to choose the best gateway for mobile IoT devices, that takes into account factors like the distance between candidate gateways and IoT devices as well as the resources of candidate gateways like processing power, memory, and bandwidth. We simulate MFC environment using MATLAB and execute the MCDM algorithm using a case study of smart traffic management in an underwater surveillance system. The results of MCDM are compared with random gateway selection algorithm. The obtained results show that our proposed approach performs 59% better in overall packet transmission and 80% percent better network lifetimes.

Keywords- IoT Gateway, Mobile Fog Computing, Multi-criteria decision making, Weighted sum model, Weighted product model, Underwater sensor network

I. INTRODUCTION

The Internet of Things (IoT) ecosystem corresponds to a broad environment that consists of connected things and technologies [1] to achieve a specific goal. The word things here means insensate entities consisting of computing elements, sensors, actuators etc. IoT has made things smart by embedding sensing, processing and communication abilities in them. The continuing advancement in the field of hardware and communication technology is improving and expanding the applications of IoT. Currently, it is used in numerous areas such as home, healthcare, industry, vehicle, animal tracking, ubiquitous and many real-time applications. The Internet of Everything (IoE) ecosystem is the enhancement of the IoT ecosystem that corresponds to the seamless interconnection and autonomous coordination of a large number of people, processes, data, and things through various heterogeneous networks. Currently, IoE systems follow the architecture of Cloud-centric Internet of Things (CIoT) for storage, analytics, and processing [2]. CIoT consists of three different layers, Embedded system layer, middleware, and Cloud data centers [3]. The embedded system layer consists of many embedded devices that interact with the environment, while the middleware layer connects this embedded system layer to the cloud system. IoE usually use Sense Process Actuate Model (SPAM) in which sensors sense, and collect data which is transmitted to the cloud through some intermediate devices such as modems, routers, and switches that act as gateways. Cloud processes this data and after processing, the results are sent back to actuators through gateways for taking necessary actions. The number of IoT devices is increasing day by day due to the development of 5G/6G technologies and Ubiquitous Computing. As per IoT Market research information, there are about 7 billion internet-connected devices already in use [4]. As internet usage increases and fresh appliances and equipment reach the market, their quantity is expected to rise in the coming years in various fields like processing and manufacturing plants, stores, clinics, universities and many more. It is anticipated that the quantity of information produced by linked Internet of Things (IoT) devices, projected to expand to 41.6 billion by 2025, will produce 79.4 zettabytes (ZB) of information [5]. International Data Corporation (IDC) estimates the quantity of information generated by linked devices to rise at an average yearly growth speed of 28.7 percent over the prediction phases for 2018-2025 [6]. Therefore in the future Cloud servers will not be capable of efficiently handling the vast amount of generated data. Furthermore, Cloud servers are multihop away from IoT devices which results in increased bandwidth consumption, network congestion and unnecessary delays in the provision of IoT services [7]. Many latency-sensitive applications or critical applications like health monitoring, can't afford these delays. To address these limitations, several approaches like Mist Computing, Mobile Computing, Edge Computing, Fog Computing, etc., have been proposed that provide processing, storage, and analytic services near the end-user application [8]. Among these approaches, Fog Computing is the one that has recently gained the most attention by both industry and academics. Fog computing is a distributed paradigm that extends Cloud computing by providing computing, networking, storage, and analytics services to the edge of the network [9]. The multi-layer bi-directional, and decentralized architecture of Fog Computing is shown in Fig. 1 which consists of three tiers.

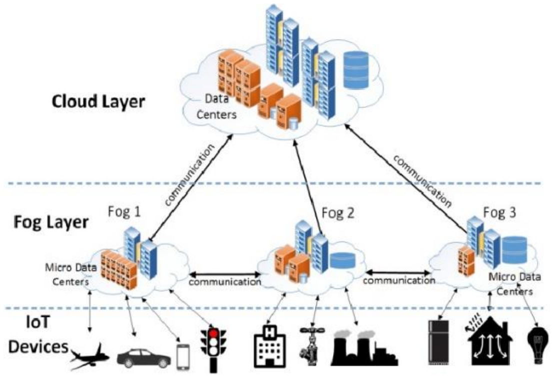

Figure 1 Fog Computing Architecture

The topmost layer is the Cloud layer, the intermediate layers consist of Fog devices, and the bottom-most IoE layer consists of sensors and actuators. The cloud layer includes a remote centralized cloud responsible for performing longterm analysis and decision-making. The middle layer consists of heterogeneous fog devices with limited computing, storage, and networking capability e.g., hubs, switches, routers, proxy servers, and radio towers. The IoT device layer consists of sensors and actuators. Sensors gather data from the environment and transmit it to Fog nodes for processing. After processing, Fog nodes send the results back to actuators to take action accordingly.

Mobile Fog Computing
Fog computing supports a variety of services like computing, storage, networking, decision-making, and mobility support that help in reducing latency, network bandwidth usage, and energy consumption [10]. A static fog computing approach is suitable, only when the end device users are at a fixed place like in a factory, at home, office, or in hospital security and authentication. The users who are moving and changing their position continuously give rise to the concept of mobile IoT devices. These mobile IoT devices led to the development of mobility-aware IoT applications called mobile IoT applications. These mobile IoT applications are real-time, latencysensitive that need an immediate response that cannot be provided by the cloud as cloud data centers are multi-hops away. Furthermore, the ever-increasing number of mobile IoT devices produces a tremendous amount of mobile big data that cannot be handled by the cloud alone. Also, the increased use of mobile objects, such as vehicles, ships, boats, drones, smartphones, tablets, etc. resulted in an increasing number of mobile IoT applications like vehicular and marine applications. These mobile IoT applications need connectivity between IoT infrastructure and mobile devices therefore to fulfill the needs of these applications the concept of Mobile Fog Computing (MFC) is introduced. MFC is the dynamic version of fog computing that emphasizes the moving user's behaviour due to having out-ofcoverage services of fog nodes. The mobile fog computing framework expands the centralized cloud architecture and 4 provides the computation closer to the users even when they are continuously altering their location [11]. It is a subset of all the parameters involved in fog computing that shows mobility in fog [12]. Mobile fog computing is defined as "a term used to explain the fog-enabled mobile IoT applications providing fast delivery, with low latency, versatility and sustainability, extra efficiency, and overall performance". Unlike the static fog computing architecture where only smart homes, smart buildings, smart factories, and production units that applied the fog computing frameworks [13], the MFC has a different perspective. MFC supports a large number of mobile fog application domains like land vehicular fog, marine fog, unmanned aerial vehicular fog, and video crowd-sourcing etc. We present two major application domains as follows:

Vehicular Fog
The fusion of fog computing and vehicular networks forms a new paradigm named Vehicular Fog Computing (VFC) for handling the mobility complexity of the congested vehicular system to

minimize the latency and quality of service [14][15]. VFC presented is an important use-case of MFC in which continuously moving vehicle mobility patterns are utilized to perform resource management on fog nodes with time-critical networking applications [16]. CognitiveRadio-based assessed protocol and distance-based forwarding protocol are used to manage the computational energy and power management and use all the available fog resources in a distributed system for effective data analysis. The network performance is enhanced by utilizing the MAC media access layer protocol in VFC and V2V communication.

II. INTERNET OF UNDERWATER THINGS

Internet of Underwater Things (IoUT)[17] is another important category of the Internet of Things, and is described as the network of intelligent aquatic objects that are interlinked. As aquatic objects like submarines are mobile IoT objects, therefore, IoUT is anticipated as one of the potential case studies of MFC applications that allow multiple mobile IoT applications, like tracking the underwater environment, exploration, and avoidance of disasters. IoUT is a prospective technique for creating smart cities with such applications. MFC provides advantages like rapidness, ultra-low latency, substitutability and sustainability, efficiency, and selfawareness to mobile IoT [12]. The dynamic MFC environment, heterogeneous network resources, heavy workloads, and the mobility of the participating objects raises many challenges. One of these challenges is to select the appropriate gateway that pre-processes huge data generated by mobile IoT devices locally before it is sent to the cloud. After pre-processing the consolidated, compiled and strategically evaluated data is transmitted to the cloud which reduces the network traffic and has a major effect on reaction times and channel delivery expenses. Moreover, the gateway provides extra safety and security to IoT networks. As these gateways handle information flowing in both directions, they can safeguard data moving to the cloud from leaks and IoT devices from being damaged by malicious external assaults with characteristics such as manipulation, encryption, detection, and random number generators for hardware and crypto motors. A mobile IoT device may have many gateway devices available in a certain area therefore, the selection of the right gateway is an important job because of its overall effect on the fog and Cloud architecture and their performance in terms of energy consumption, resource, and network usage. In MFC environment, the mobility of devices makes the optimal gateway selection more challenging. Some of the required features for IoT gateway devices are data aggregation, data buffering, data preprocessing, data streaming, data cleansing, filtering, data security, device configuration management and system fault tolerance and rehabilitation [18]. All these features and requirements stress the importance of optimal gateway selection. The existing algorithms consider only one criterion of the minimum distance for gateway selection which is called Random gateway selection. So, there is a need to develop an optimal gateway selection algorithm that can improve packet transmission and network lifetime.

Contribution and goal of the paper

In this paper, we propose a novel multi-criteria decision-making (MCDM) algorithm for optimal gateway selection for mobile IoT devices used in underwater surveillance systems. The system is radar reflector-based and spread across a coastal region and the objective is to safeguard shipping lanes by detecting items such as submarines. As the field has many of gateway nodes capable of transmitting off-field via a satellite or aircraft. Therefore effective gate selection is very critical in such a system to improve overall packet transmission, throughput, and network lifetime. The major contributions of the proposed work are as follows:

- We provide the architectural model for an underwater surveillance system in a Mobile Fog Computing environment, where a self-configured network is established between heterogeneous sensor nodes and level-1 gateway devices. The level-1 gateway devices send data after aggregation to level-2 gateway devices that transfer it to the cloud.
- We propose a novel multi-criteria decision-making (MCDM) algorithm for optimal gateway selection for mobile IoT devices used in underwater surveillance systems. Our algorithm selects the gateway on four criteria: distance of various gateway devices from IoT devices, available resources of the gateway devices that include memory, bandwidth, and processing of the gateway.
- To simulate the MFC environment and implement the MCDM algorithm we use MATLAB and the obtained results reflect that our proposed scheme produces better results for overall packet transmission and network lifetime.

This paper is organized as follows: the section 3 presents the background. Section 4 presents the architectural model, and proposed scheme for optimal gateway selection by using MCDM. Section 5 discusses the performance evaluation and obtained results and, finally, we conclude in Section 6.

III. RELATED WORK

The development of an effective IoT system is a difficult task because of the following main problems. Firstly, the chosen sensor network technology must be resource efficient and ideal for IoT applications. In some IoT applications sensor nodes, their memory, processing capabilities, communication speed, and power are considerably reduced as compared to sensors in other network areas. Secondly, IoT systems often need to handle streaming-based communications when real-time requirements need to be met, unlike traditional sensor networks with interval-based data (e.g. temperature and moisture monitoring). Thirdly, hardware systems with high computing power and parallel processing (e.g., multi-core processors), due to the concurrent existence of working loads, are required on the gateway in multi-patient applications like smart hospitals [19]. Issam Jabri and fellows propose a fuzzy logic with an ant colony optimization-based approach for vehicular gateway selection on the Internet of Vehicles (IoV) [20]. The proposed scheme has two phases, Firstly a set of candidate gateway nodes are selected using Fuzzy Logic and secondly, the number of selected gateways are optimized. To solve the multi-optimization problem the authors apply an ant colony optimization algorithm. The simulation results reveal that the proposed approach is more efficient in terms of reducing the number of selected gateways. Maiti et al. apply five techniques based on randomized, greedy, kMeadian, kMeans and simulated annealing to minimize delay for gateway selection among a set of fog nodes [21]. The obtained results suggest that simulated annealing gives the minimum delay. The capability of end devices in computing grows, and Fog computing has brought storage and computing nearer to the edge of the network [22]. The edge devices can pre-process raw data which helps in greatly reducing the data to be transmitted to the cloud. Also storing statistics and analyzing data at higher levels on remote servers is more secure. However, the disadvantage of more processing in the network is that download and display of data are not sufficient to show data at a greater cost for greater security. Now the days three [23] different implantation of IoT devices are available. In cloud-enabled IoT solutions devices are usually linked to the cloud by using a gateway or linking to several IoT networks, while in fog-enabled IoT solutions, to minimize response time between the IoT and the IoT network devices are connected to a local server located closer to the machine. In the third category edge enabled solutions the intelligence is passed to the network's end devices. Calculation and communication between devices and the cloud are provided by the IoT gateway [24][25]. This architecture is suitable for IoT (smart home, home health, etc) smaller solutions. The computing power of the end devices is improved and so intelligent algorithms can be implemented. Devices are rapidly advanced; they must have following capabilities like minimization of the network traffic, improve system response time, better data security over transmission.

Quy suggested a gateway selection strategy for MANET-IoT that takes responsiveness, queue length, and gateway distance into account [26]. The simulation results show a significant improvement in load balancing, packet delivery ratio, latency, and other performance metrics.

The authors proposed a fuzzy-based hybrid multi-criteria decision-making optimisation technique for gateway selection based on subjective weights derived from the Fuzzy Best-Worst Method (Fuzzy-BWM) and Fuzzy Level Weight Assessment (Fuzzy-LBWA) to synthesise nonlinear weights [27]. They use Vector-normalized Fuzzy Combined Compromise Solution with Later Defuzzification (V-Fuzzy-CoCoSo-LD) methods to rank nodes based on Quality of Service and Level of Security. The study found that the vector normalisation method works best for the Fuzzy-CoCoSo MCDM model, with mean Spearman's and Pearson's correlation coefficients of 0.9404 and 0.9868, respectively.

IV. Design and Implementation

In this Section, we present an architectural model, an example case study, our proposed MCDM-based gateway selection scheme, algorithm and flow chart for MCDM.

Case Study

The underwater surveillance system is an important case study for MFC. Here we discuss the case study for smart traffic management in an underwater surveillance system. A smart traffic management system contains a set of rivulet queries running on data produced by sensors deployed under the water. In terrestrial wireless networks normally we use radio frequency (RF) to establish a network, while underwater due to water absorption; these radio networks do not work well. As sound has better propagation characteristics so underwater network normally uses acoustic signals. There are numerous factors that affect the design process for underwater surveillance systems like limited Bandwidth, propagation delay, shadow zones, limited energy, sensor node failure, and attenuation [28]. Such kinds of networks are designed for either ocean-bottom monitoring or ocean-column monitoring, or detection of autonomous underwater vehicles. Similarly, sensor node deployment is a challenging task. Few networks are designed for longterm full-time-Critical Aquatic Monitoring and few others for short-term part-time-Critical Aquatic exploration [29].

Architectural Model

As shown in Fig. 3 below, a self-configured network is established between heterogeneous sensor nodes and a few level-1 gateway devices. Sensor nodes sense data from the environment and transfer this data to level 1 gateway fog devices. While level-1 devices aggregate this data and send it to level-2 devices. From level-2 fog devices data has been transferred over to the cloud network for data analysis and appropriate action. Level-1 and level-2 devices are used as gateways. In our experiment, we propose a new gateway selection mechanism by which network lifetime, data throughput, and network stability can be improved.

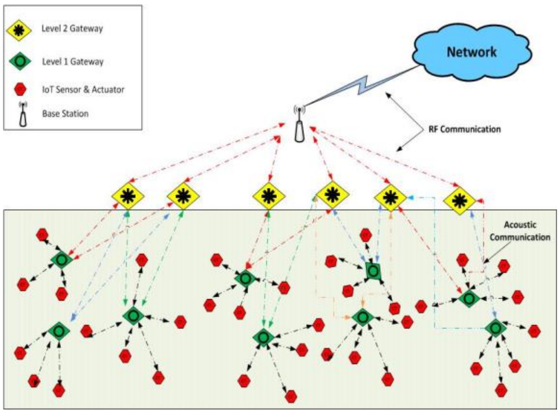

Figure 2 A case study for proposed scheme, underwater sensor network (UWSN) Under the water acoustic communication while RF Communication between Level-2 fog gateway and cloud networks

The most of design and application of gateway selection are in many scientific articles based on edge-enabled solutions and work over single criteria, so we have decided to work on multi-criteria decision-making for gateway selection. The detail of this technique is discussed in the following section.

Multi-criteria decision-making (MCDM) based gateway selection

Multi-criteria decision-making (MCDM) is a technique that handles with decisions making for the selection of the best option from several possible candidates in a decision, subject to multiple criteria or attributes that may be concrete or vague. It deals with the selection of alternatives without prior knowledge of the best. Here we handle the 4 different criteria of many gateways, i.e. distance from node, available bandwidth, available memory and MIPS and select a single gateway that can provide the best feature services to the IoT nodes. According to our proposed scheme, an IoT node that wishes to communicate data first broadcast a hello message to its neighbour, and in response all available gateways send their response along with their features. The IoT node runs the MCDM technique and selects the most optimal gateway for the transmission of its data.// The optimal gateway selection may increase the throughput by improving the packet delivery ratio, and network lifetime. Here we proposed the gateway selection scheme for fog networks by using the MCDM technique. We build the 4x4 matrix on the value of the distance of the gateway from the IoT device, available memory, bandwidth and MIPS of the gateway. We denote these values by alpha (α), beta (β), gamma (γ) and delta (δ) respectively. We get the distance as a variable because as the distance increases between the gateway and IoT devices, the transmission of data affects negatively. Many times IoT devices have limited energy resources and most of the energy is consumed during the transmission of data. The second value in the matrix is available in the memory of gateway devices. As memory is one of the main resources utilized during processing and different operations. Gateway devices that have more memory available, may process the IoT data earlier. The third value in the matrix is available bandwidth. The gateway devices that have more bandwidth may provide faster transfer speed and latency issues may be resolved. The last value matrix is MIPS, which is actually the speed of the processor. A fog gateway device may have more MIPS available and can give an earlier and quick response as compared to a device that has low MIPS available. In the gateway selection mechanism, the IoT device that wants to transmit the signal will broadcast its HELLO message, this message is received by various candidate gateway devices, and the gateway will send the HELLO_REPLY message. Every gateway device will share its parameter with IoT devices in this response. After that first, we calculate the decision matrices by using the weighted sum model (WSM) for every gateway candidate node. WSM is one of Multi-Criteria Decision Making (MCDM. Here we have multiple attributes and each of them has a different value range and different units so in this method, we assign the weight according to importance level. The following Equation (1) is used for matrix values:

$$X_{ij} = n_2 \begin{bmatrix} a & \beta & \gamma & \delta \\ n_i & x_{12} & x_{13} & x_{14} \\ \vdots & X_{22} & x_{23} & x_{24} \\ \vdots & \vdots & \vdots & \vdots \\ x_{i1} & x_{i2} & x_{i3} & x_{j4} \end{bmatrix} \quad (1)$$

So this technique provides us with quality ranking (Q) of our matrices. WSM computes the overall score of a metric as the weighted sum.WSM work under three steps:

1. Scaling the equivalent value (Scaling value may obtain on positive or negative criteria). Distance is a negative criterion because it should be smaller between IoT and gateway devices, while memory, Bandwidth, and MIPS are the positive criteria because their larger value is better. The best and worst score is between 1 and 0.

$$Negative\ Criterion = \frac{\mathrm{Min}(x_{ij})}{x_{ij}} \quad (2)$$

$$Positive\ Criterion\ = \frac{x_{ij}}{\mathrm{Max}(x_{ij})} \tag{3}$$

While..... $X\mathcal{E}[0,1]$ (4)

2. After calculating the comparable value for each X_{mn}, we assign the weighting factor to specify the priority value. For priority calculation, there are two methods, eigenvector and direct specification. Here we use the direct specification method and in this scenario, we consider the equal priority for all variables. Below is the equation of the weighting factor where ωj is the weighting factor and m is the number of matrices. While ωα, ωβ, ωγ, and ωδ are the weighting factors for distance, memory bandwidth, and MIPS respectively.
   Weighting Factor, ... $\sum_{j=1}^{m} \omega_j = 1$ (5)
   Whereas
   $\omega_j = \omega_\alpha + \omega_\beta + \omega_\gamma + \omega_6$ (6)
   In the last step of WSM we sum up all the weight values of each metric. So this technique provides a first-quality ranks of all available gateways.

   $$Q_i^1 = W^{wsm} = \sum_{j=1}^{m} \omega_j X_{ij} \tag{7}$$

   After calculating Q_i^1 through WSM technique, we calculate Q_i^2 by using the weighted product model (WPM). In the WPM model first two steps are the same as in the WSM model. So we utilize the same values generated by equation (2) to equation (6). While in the last step, the weighting factor is used in the power of scaling value. The equation for the WPM model is:

   $$Q_i^2 = W^{\omega\rho m} = \prod_{j=1}^{n} x_{ij}^{\omega_j} \tag{8}$$

3. In the last step of this algorithm now we calculate the Q_i by using the results of equation (7) and equation (8) with the following equation.
   $Q_i = \lambda Q_i^1 + (1-\lambda) Q_i^2$ (9)

   Here the optimal λ values for each of the considered problems are determined and the effects of varying λ values on the ranking of the candidate alternatives. Value of λ may be between 0 and 1. If the value of λ is 0, the total result may shift to the WPM model and if the value of λ is 1, the entire result will be shifted to the SPM model. In our example, we consider the value of λ as 0.5

MCDM Algorithm and Flow chart

Edge and fog gateway nodes in underwater surveillance systems play a pivotal role in preprocessing, executing aggregation and compression of data from multiple sensors, which minimizes latency and improves bandwidth utilization.

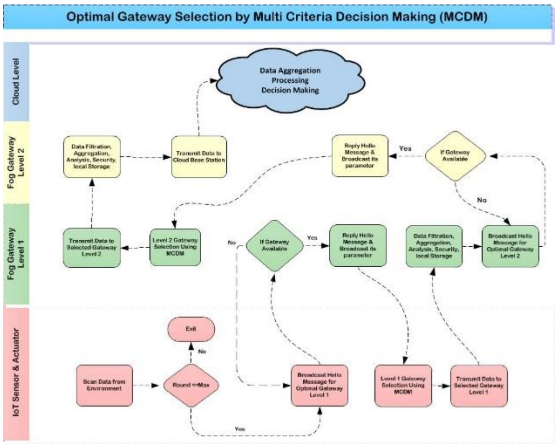

Figure 3 Flow chart for Gateway selection mechanism by using Multi-Criteria Decision Making

Fig. 3 presents the whole process of gateway selection that has been subdivided into 4 layers. At the first lower layer, IoT sensors have been used for three basic tasks, i.e. sense information from the environment, data transmission, and selection of optimal gateway level-1 by using the MCDM algorithm. At the second layer, fog gateway level-1 devices have been used for the functionality of data transmission, data aggregation and level-2 gateway selection by using the MCDM algorithm. At the layer-3 fog gateway, level-2 devices have been used. The first two levels used acoustic communication only as they are under the water. Layer 3 devices have dual functionality, they communicate with layer-2 with acoustic communication while with cloud layer-4 they use RF communication. Layer-3 devices are also used for data preprocessing that includes data aggregation and data transmission. Furthermore, these gateway nodes store data packets locally and delete them after confirming receipt. This deletion of information only after confirmed receipt reduces the risk of sensitive data loss during transmission by ensuring data integrity and preventing unauthorized access that boisters system reliability. Layer-4 is a cloud network, where data aggregation, data analysis, decision-making, and data storage are performed. The variables and their symbols used in the algorithm and flowchart are given in Table 1.

Table 1. Variables Used in Algorithm

Parameters	Symbol Used
IoT Devices	D
Fog Level 1 Gateway	G1
Fog Level 2 Gateway	G2
Communication Round	R
Distance of gateway from IoT device	R
Available Ram	B
Available Bandwidth	γ
Available MIPS	δ
Selected Gateway	S
Cloud Level	C

Algorithm1: Gateway Selection Algorithm

Algorithm 1 Gateway Selection Algorithm

```
D Scan for data
if Data available to Transmit then
    for R < = R_Max do
        D broadcast Hello Msg to all G1 in the region
        do
            if G1 available then
                G1 reply with (α, β, γ, δ)
                S= MCDM (α, β, γ, δ) at D
                Send data D to G1(S)
                Data Prepossessing at G1
                G1 broadcast Hello Msg all G2 in region
                do
                    if G2 available then
                        G2 reply with (α, β, γ, δ)
                        S= MCDM (α, β, γ, δ) at G1
                        Send data to G2(S)
                        Data Preprocessing at G2
                        G2 Send data to C
                    else
                        G1 broadcast Hello Msg
                    end if
                while Data to be Send from G1
            else
                D broadcast Hello Msg to all G in region
            end if
        while Data to be Send from D
    end for
end if
```

Algorithm 2 presents the proposed MSDM algorithm for gateway selection algorithm as given below.

Algorithm 2: MSDM Algorithm

Algorithm 2 MSDM Algorithm

```
INPUT: m=Col, n=Rows, Q=Quality rank, MCDM (α, β, γ, δ)
OUTPUT: Q
Maintain matrix X_ij with equation 1
Calculate -ve and +ve criterion feature with equation 2 & 3
Assign an equal weight to all features
Wsum=0; Wpro=1;
for m=1; m<=4; m++ do
    Q1 = sum + (weight × X_ij)
end for
for n=1; n<=4; n++ do
    Q2= Wpro × (X_ij)^weight
end for
Q= λQ1+(1-λ)Q2
return Q
```

V. Performance Evaluation

We use MATLAB R2009a tool for the simulation of Optimize Gateway Selection by using Multi-Criteria Decision Making (MCDM) technique in fog computing. We use a field with dimensions 100m ∗ 100m. The total number of IoT devices n = 200. Normal and gateway level nodes are deployed in the underwater field randomly, while level-2 gateway devices are used over the surface of the water. There are 50 level-1 gateway and 5 level-2 gateway devices are used. All these devices are heterogeneous in terms of energy level and are deployed randomly in the environment. In our simulation, we use a packet size of 400 bits. Level-1 gateway and level-2 gateway both perform aggregation and compress the data packet before transferring forward. They also save these packets locally up to acknowledgement received from the next device. This acknowledgement-based deletion approach adds a layer of security that makes the whole system reliable and improves the overall performance of the system. Rest of the parameters used in the simulation are as under:

Table 2. Pamemoryeters for simulation on MATLAB

Parameter	**Value**
Field Dimension	100m x 100 m
Total IoT devices n	n=200
Level 1Gateway Devices	g=50
Level 2 Gateway Devices	G=5
Deployment of IoT and Gateway devices	Randomly
Initial Energy of Gateway devices	Eog=10 Jouls
Initial Energy of IoT devices	Eo= 2 Joules
Packet Size	4000 bit
Transmitter/Receiver Electronics	Eelec= 50e-9 , 50 nJ/bit
Data Aggregation	EDA = 5e-9 , 5 nJ/bit/Message

As we run our code with the parameter given above, the IoT devices and gateway nodes randomly deployed in the given field area as shown in the Fig. 4.

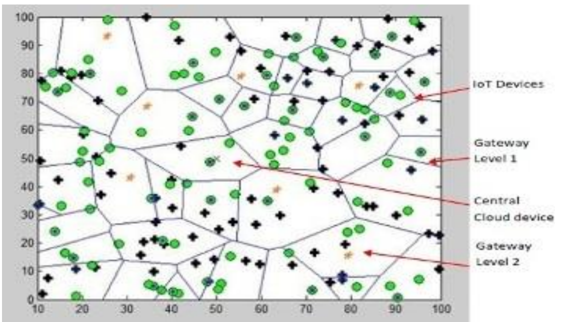

Figure 4 IoT and Gateway devices deployed in area

IoT devices that want to transmit the data, will broadcast the “HELLO” message in the region. The available gateway devices that receive this message, will reply to the “REPLY_HELLO” message. This reply contains the 4 parameters of every gateway device. These parameters are the distance of every gateway device with IoT device, available resources that include memory, bandwidth and MIPS along with gateway ID. Once the IoT device receives these parameters, it will use MCDM to calculate the optimum gateway. Then the device will send its data to that gateway

Gateway selection by MCDM an Example

As we describe 200 IoT devices, 50 level 1 gateway devices and 5 level 2 devices were used in the simulation experiment. Simulation experiment done for 1000 rounds. How level-1 and level-2 gateways have been selected is shown in the following example where parameters received by a particular IoT device are shown here in the following table. Every IoT device establishes a parameter matrix in its cache according to equation (1).

Table 3. IoT received data from different gateways

Variables→	**Distance(α)**	**memory (β)**	**Bandwidth (γ)**	**MIPS(δ)**
Gateway Devices↓				
Gateway 1	10	512 MB	1000	250
Gateway 2	8	512 MB	2000	500
Gateway 3	12	1024 MB	1000	250
Gateway 4	2	1024 MB	1200	500
Gateway 5	9	512 MB	1500	1000

Table 4. Equivalent values with Positive and Negative Criterion

Weight	**0.25**	**0.25**	**0.25**	**0.25**
Variables→	Distance(α)	memory (β)	Bandwidth (γ)	MIPS(δ)
Gateway Devices↓				
Gateway 1	0.2	0.5	0.5	0.25
Gateway 2	0.25	0.5	1	0.5
Gateway 3	0.166	1	0.5	0.25
Gateway 4	1	1	0.6	0.5
Gateway 5	0.22	0.5	0.75	1

Once the parameters matrix is saved, IoT device calculates the equivalent scaling value according to positive and negative criteria with the help of equations
(2) and (3). The following results are calculated:

Table 5. Weighted Equivalent values of X_{ij}

Variables→	**Distance(α)**	**memory (β)**	**Bandwidth (γ)**	**MIPS(δ)**
Gateway Devices↓				
Gateway 1	0.05	0.13	0.13	0.06
Gateway 2	0.06	0.13	0.25	0.13
Gateway 3	0.04	0.25	0.13	0.03
Gateway 4	0.25	0.25	0.15	0.13
Gateway 5	0.06	0.13	0.19	0.25

After calculating the equivalent value, weights are assigned to each column and multiply them with each X_{ij} . In our case, we assign equal weight to all parameters as discussed in equation (6). This is also to be noted that the sum of all weights is equal to 1, as given in equation (5).

Table 5. Weighted Equivalent values of X_{ij}

Weight	**0.25**	**0.25**	**0.25**	**0.25**
Variables→	Distance (α)	memory (β)	Bandwidth (γ)	MIPS(δ)
Gateway Devices↓				
Gateway 1	0.05	0.13	0.13	0.06
Gateway 2	0.06	0.13	0.25	0.13
Gateway 3	0.04	0.25	0.13	0.06
Gateway 4	0.25	0.25	0.15	0.13
Gateway 5	0.06	0.13	0.19	0.25

At the end of the first iteration, the value of $Q_i^1$is calculated by using equation (7). IoT devices save these values as first-quality parameters.

Table 6. Calculated value of Q_i^1

Variables →	**Distance(α)**	**memory (β)**	**Bandwidth (γ)**	**MIPS (δ)**	Q_i^1
Gateway Devices↓					
Gateway 1	0.05	0.13	0.13	0.06	0.36
Gateway 2	0.06	0.13	0.25	0.13	0.56
Gateway 3	0.04	0.25	0.13	0.03	0.48
Gateway 4	0.25	0.25	0.15	0.13	0.78
Gateway 5	0.06	0.13	0.19	0.25	0.62

In the second round, the IoT devices calculate the Q_i^2 by using equation (8) and the results are as under:

Table 7. Calculated value of Q_i^1

Variables→	**Distance(α)**	**memory (β)**	**Bandwidth (γ)**	**MIPS (δ)**	Q_i^1
Gateway Devices↓					
Gateway 1	0.67	0.84	0.84	0.71	0.33
Gateway 2	0.71	0.84	1.00	0.84	0.56
Gateway 3	0.64	1.00	0.84	0.71	0.38
Gateway 4	1.00	1.00	0.88	0.84	0.74
Gateway 5	0.68	0.84	0.93	1.00	0.54

In the last step of this algorithm, the IoT devices calculate the Qi by using the results of equation (7) and equation (8) with equation (9). Here the λ is considered as 0.5. The highest value of Qi is achieved at 0.76 so corresponding gateway 4 has been selected by the MCDM algorithm for data transmission.

Table 8. Calculated value of Q_i

WSM, Q_i^1	**WPM, Q_i^2**	**Q_i , λ=0.5**
0.36	0.33	0.35
0.56	0.50	0.53
0.48	0.38	0.43
0.78	0.74	0.76
0.62	0.54	0.58

VI. Results and Discussion

IoT sensing devices selects the optimal gateway with the same methodology in each cluster as shown in Fig. 5. Once gateways have been selected, sensing devices start data transferring to these gateways. Every gateway aggregates this data and then transfers it to the cloud central device. This procedure continues in every round that is predecided in our experiment. Here we run this up to 2000 rounds. After execution of a few rounds, the batteries of some sensor nodes dried and their status is represented as a dead node, as shown in Fig. 6 below.

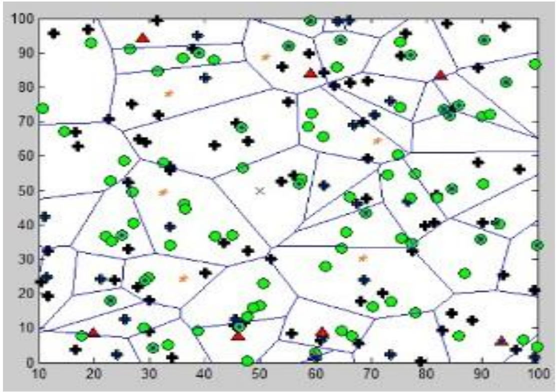

Figure 5 Dead nodes appears as their batteries dried (Dead node in red color)

In the simulation, we repeated this experiment twice. In the first experiment, we selected gateways by using our proposed method of Optimal gateway selection through multi-criteria decision-making (OGS-MCDM). In the second experiment, the gateways are selected through a random gateway selection (Random GS) mechanism over the same parameters and environment.

Performance Metrics

To evaluate the efficiency of the proposed OGS_MCDM, we used the following performance metrics.

Network Throughput

Network throughput is measured by the summation of all packets transferred to the cloud through selected gateways as shown in the following equation:

$$P_{TRN} = \sum_{i=1}^{N} T_i \quad (10)$$

Network Stability: We have compared network lifetime in terms of the number of alive nodes and dead nodes with respect to the number of rounds. The "alive" gateway nodes refer to network devices such as computers, routers, switches, servers, and other edge and fog devices that are up, running and responsive, ensuring consistent communication. On the other hand, "dead" nodes are network devices that are offline, unresponsive, and unable to send and receive data. The count of alive and dead nodes indicates the health and well-functioning of the network and the application of various network management and resource management techniques such as load balancing, capacity planning and resource allocation. This continuous monitoring of the status of alive and dead nodes provides a comprehensive analysis of network performance and improves the security and reliability of the system along with optimization of system performance. We have compared compared our proposed MCDM with Random Gateway Selection algorithm using three criteria i.e. number of packets transferred to the central cloud node, dead nodes and alive nodes.

Table 9 shows the number of packets transferred using our proposed scheme OGS-MCDM and Random GS algorithm according to simulation results.

Table 9. Number of Packets Transferred

Algorithm	**Packets Transferred**
OGS-MCDM	28K
Random GS	16K

Table 9 shows that number of packets transferred using our proposed scheme OGS-MCDM are much greater than Random GS from every cluster gateway to the central base station (Cloud Network station). It means the throughput of the network is improved if we use the appropriate gateway to forward data from the sensor node to the cloud. These statistics show our results are almost 59 percent better than the random gateway selection mechanism.

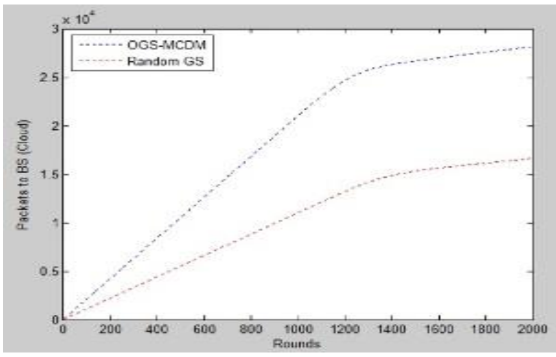

Figure 6 Comparison of two techniques in terms of packets transferred from sensor node to cloud.

Fig. 6 presents the comparison results in terms of packet transferred using both techniques. The rounds are presented along x-axis while number of packers transferred are shown along y-axis. Number of packets transferred using OGS-MCDM and, Random GS are shown with blue and purple colors respectively.

Table 10 shows the comparison of first dead node appearance using both algorithms.

Table 10. First dead node appearance

Algorithm	**Round No**
OGS-MCDM	968
Random GS	1147

The table shows that in our proposed scheme appearance of the first dead node is earlier than the Random gateway method.

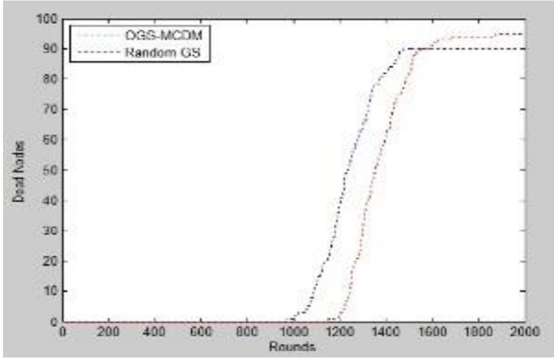

Figure 7 Comparison of two techniques in terms of dead node appeared

Fig. 7 presents the comparison graph for two techniques in terms of dead node appearance where MCDM and Random GS results are represented with blue color and purple color respectively.
In later results, it proves that after a certain node dead our methodology shows better results as it is more stable than the random gateway selection method. Table 11 shows the total number of alive nodes at the end of 2000 round using our proposed algorithm and RGS.

Table 11. Total Number of Alive Nodes

Algorithm	Number of Alive Nodes
OGS-MCDM	10
Random GS	2

Table 11 shows that at the end of 2000 rounds in our proposed scheme about 10 sensor nodes were alive while there is only 2 nodes are alive in the random gateway method.
Fig. 8 presents the comparison graph for two techniques in terms of number of alive nodes after round number 2000. The number of alive nodes using MCDM and Random GS results are represented with blue color and purple color respectively.

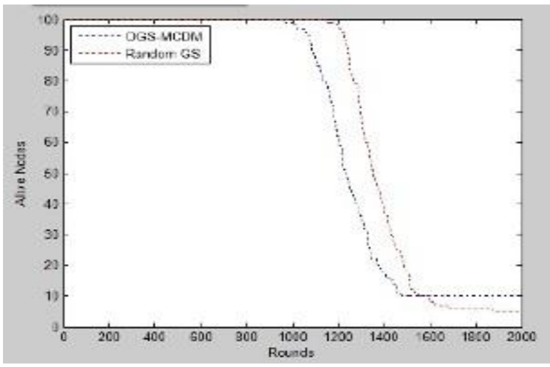

Figure 8 Comparison of two techniques in terms of alive node and network stability

VII. Conclusion

Intelligent gateways in the presence of submarine vehicle sensor nodes will exploit their specific strategic position in IoT systems to address several challenges, including mobility, electricity and resource saving, inter-operability, scalability and reliability. In this paper, we select optimize gateway by using Multi-Criteria Decision Making (MCDM) technique in Fog Computing. Here we used Weighted Sum Model (WSM) and Weighted Product Model to calculate the quality parameters of available different gateways. Then we find the final Q value by using a weighted aggregated sum product assessment. Our model results prove that our proposed method improves the network lifetime, its stability period and the Packet-Delivery-Ratio (PDR) of cloud network. In future research activities, we aim to expand the existing environment in order to promote proximity-conscious data management and the provision of adaptive services with the mobility of IoT underwater sensor nodes.

References

[1] Laghari, A. A., Wu, K., Laghari, R. A., Ali, M., & Khan, A. A. (2021). A review and state of art of Internet of Things (IoT). Archives of Computational Methods in Engineering, 1-19.

[2] Alhaidari, F., Rahman, A., & Zagrouba, R. (2023). Cloud of Things: architecture, applications and challenges. Journal of Ambient Intelligence and Humanized Computing, 14(5), 5957-5975.

[3] Verma, N., Singh, S., & Prasad, D. (2022). A review on existing IoT architecture and communication protocols used in healthcare monitoring system. Journal of The Institution of Engineers (India): Series B, 103(1), 245-257.

[4] Sami, N., Mufti, T., Sohail, S. S., Siddiqui, J., Kumar, D., & Neha. (2020). Future Internet of Things (IOT) from Cloud perspective: Aspects, applications and challenges. Internet of Things (IoT) Concepts and Applications, 515-532.

[5] Alam, T. (2018). A reliable communication framework and its use in internet of things (IoT). CSEIT1835111| Received, 10, 450–456.

[6] Al-Amiedy, T. A., Anbar, M., Belaton, B., Kabla, A. H. H., Hasbullah, I. H., & Alashhab, Z. R. (2022). A systematic literature review on machine and deep learning approaches for detecting attacks in RPL-based 6LoWPAN of internet of things. Sensors, 22(9), 3400.

[7] Jamil, B., Ijaz, H., Shojafar, M., Munir, K., & Buyya, R. (2022). Resource allocation and task scheduling in fog computing and internet of everything environments: A taxonomy,

review, and future directions. ACM Computing Surveys (CSUR), 54(11s), 1-38.

[8] Hazra, A., Rana, P., Adhikari, M., & Amgoth, T. (2023). Fog computing for next-generation internet of things: fundamental, state-of-the-art and research challenges. Computer Science Review, 48, 100549.

[9] Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., et al. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. Journal of Systems Architecture.

[10] Jamil, B., Ijaz, H., Shojafar, M., Munir, K., & Buyya, R. (2022). Resource Allocation and Task Scheduling in Fog Computing and Internet of Everything Environments: A Taxonomy, Review, and Future Directions. ACM Computing Surveys (CSUR).

[11] Gao, X., Huang, X., Bian, S., Shao, Z., & Yang, Y. (2019). PORA: Predictive offloading and resource allocation in dynamic fog computing systems. IEEE Internet of Things Journal, 7(1), 72–87.

[12] Abbas, A., Khan, S. U., & Zomaya, A. Y. (2020). Fog Computing: Theory and Practice. John Wiley & Sons.

[13] Alam, M. G. R., Tun, Y. K., & Hong, C. S. (2016). Multi-agent and reinforcement learning based code offloading in mobile fog. In 2016 International Conference on Information Networking (ICOIN) IEEE; 2016. p. 285–290.

[14] Zhai, Y., Sun, W., Wu, J., Zhu, L., Shen, J., Du, X., et al. (2020). An energy aware offloading scheme for interdependent applications in software-defined IoV with fog computing architecture. IEEE Transactions on Intelligent Transportation Systems, 22(6), 3813–3823.

[15] Liu, C., Liu, K., Ren, H., Zhou, Y., Feng, L., Guo, S., et al. (2019). Enabling Safety-Critical and Computation-Intensive IoV Applications via Vehicular Fog Computing. In 2019 15th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN) IEEE; 2019. p. 378–383.

[16] Jamil, B., Ijaz, H., Shojafar, M., & Munir, K. (2023). IRATS: A DRL-based intelligent priority and deadline-aware online resource allocation and task scheduling algorithm in a vehicular fog network. Ad Hoc Networks, 103090.

[17] Kao, C. C., Lin, Y. S., Wu, G. D., & Huang, C. J. (2017). A comprehensive study on the internet of underwater things: applications, challenges, and channel models. Sensors, 17(7), 1477.

[18] Gluhak, A., Krco, S., Nati, M., Pfisterer, D., Mitton, N., & Razafindralambo, T. (2011). A survey on facilities for experimental internet of things research. IEEE Communications Magazine, 49(11), 58–67.

[19] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, 17(4), 2347–2376.

[20] Jabri, I., Mekki, T., Rachedi, A., & Jemaa, M. B. (2019). Vehicular fog gateways selection on the internet of vehicles: A fuzzy logic with ant colony optimization based approach. Ad Hoc Networks, 91, 101879.

[21] Maiti, P., Apat, H. K., Sahoo, B., & Turuk, A. K. (2019). An effective approach of latency-aware fog smart gateways deployment for IoT services. Internet of Things, 8, 100091.

[22] Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. IEEE Internet of Things Journal, 3(5), 637–646.

[23] Jan, M. A., Zhang, W., Usman, M., Tan, Z., Khan, F., & Luo, E. (2019). SmartEdge: An end-to-end encryption framework for an edge-enabled smart city application. Journal of Network and Computer Applications, 137, 1–10.

[24] Jung, E. Y., Kim, J. H., Chung, K. Y., & Park, D. K. (2013). Home health gateway based healthcare services through U-health platform. Wireless Personal Communications, 73(2), 207–218.

[25] Rahmani, A. M., Gia, T. N., Negash, B., Anzanpour, A., Azimi, I., Jiang, M, et al. (2018). Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach. Future Generation Computer Systems, 78, 641–658.

[26] Quy, V. K., Ban, N. T., Van Anh, D., Quy, N. M., & Nguyen, D. C. (2023). An adaptive gateway selection mechanism for MANET-IoT applications in 5G networks. IEEE Sensors Journal.

[27] Ogundoyin, S. O., & Kamil, I. A. (2023). An integrated Fuzzy-BWM, Fuzzy-LBWA and V-Fuzzy-CoCoSo-LD model for gateway selection in fog-bolstered Internet of Things. Applied Soft Computing, 143, 110393.

[28] Akyildiz, I. F., Pompili, D., & Melodia, T. (2005). Underwater acoustic sensor networks: research challenges. Ad hoc networks, 3(3), 257–279.

[29] Dong, B., & Mahdy, A. M. (2010). Underwater Wireless Sensor Networks: Efficient Localization Schemes using SemiDefinite Programming. International Journal on Advances in Networks and Services, 3(1 & 2), 2010.