IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 13, 2021, accepted April 28, 2021, date of publication May 13, 2021, date of current version May 24, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3080133*

# REMOS-IoT-A Relay and Mobility Scheme for Improved IoT Communication Performance

**ANDERSON AUGUSTO SIMISCUKA, (Member, IEEE),**
**AND GABRIEL-MIRO MUNTEAN, (Senior Member, IEEE)**
School of Electronic Engineering, Dublin City University, Dublin 9, D09 W6Y4 Ireland

Corresponding author: Anderson Augusto Simiscuka (anderson.simiscuka2@mail.dcu.ie)

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Program for the TRACTION Project under Grant 870610, in part by the Science Foundation Ireland (SFI) under Grant SFI/12/RC/2289_P2 (Insight) and Grant SFI/16/SP/3804 (ENABLE), and in part by the European Regional Development Fund.

**ABSTRACT** The Internet of Things (IoT) can avail from device-to-device (D2D) communication techniques to increase object data exchange performance. IoT networks aim to offer a massive number of services at high quality levels, and many of the devices providing these services are mobile. Devices such as wearables, sensors, drones and smart vehicles need constant connectivity despite their moving patterns and therefore, an IoT architecture should consider both Quality of Service (QoS) and mobility. D2D allows devices to communicate directly to share content and functionality, such as access to the Internet. This paper proposes REMOS-IoT - A RElay and MObility Scheme for improved IoT communication performance in support of increased QoS for the data exchange services between mobile IoT devices. Simulation-based testing showed how performance of devices increased in several scenarios, demonstrating the efficiency of the proposed architecture and algorithms.

**INDEX TERMS** Internet of things (IoT), smart gateways, D2D, QoS, performance analysis, mobility.

## I. INTRODUCTION

The Internet of Things (IoT) is expected to reach 18 billion interconnected devices by 2022 [1]. Based on the new paradigms brought by the evolving technology, a recent IEEE report [2] has identified important areas of research in IoT. These areas include real-time coordination, data storage, network performance, concurrency, mobility patterns, quality of service (QoS), etc. Innovative solutions can improve the values of some metrics during data delivery, especially in terms of QoS metrics, such as delay, packet loss and throughput. In order to assess and enhance QoS, solutions employ several approaches, including adaptation schemes [3], [4], scheduling techniques [5], [6], clustering algorithms and other innovative communication approaches [7], [8].

IoT devices benefit from the latest device-to-device (D2D) communication paradigm, which allows direct mobile device intercommunication, when they cannot reach any base station or there is limited bandwidth available. Other benefits of D2D include energy conservation, QoS improvements, and load balancing optimization, as certain devices relay data via other devices [9], [10]. D2D has also been used in cooperative vehicular networks [11]. Architectures that enable IoT devices use D2D communications have been proposed in [12]–[14], but they have not focused specifically on performance.

This paper proposes **the RElay and MObility Scheme for the Internet of Things (REMOS-IoT)** which enhances the existing IoT architecture with algorithms which help under-performing devices and improve mobile IoT device connectivity performance. REMOS-IoT also provides mobility and relay alternatives to devices. Its main goal is to analyze smart devices' performance metrics and based on their values provide the best network connectivity by assigning the devices to optimal IoT smart gateways in the IoT network. In this process REMOS-IoT achieves an increase of QoS for data exchange services between mobile IoT devices.

Fig. 1 illustrates the REMOS-IoT architecture and its major components and shows an example of an usage scenario in the context of a mobile urban environment. REMOS-IoT contains IoT devices (or objects), smart gateways and a cloud-based IoT Integration Platform (ITINP). IoT objects provide and consume services from other objects in the

---

The associate editor coordinating the review of this manuscript and approving it for publication was Tawfik Al-Hadhrami.

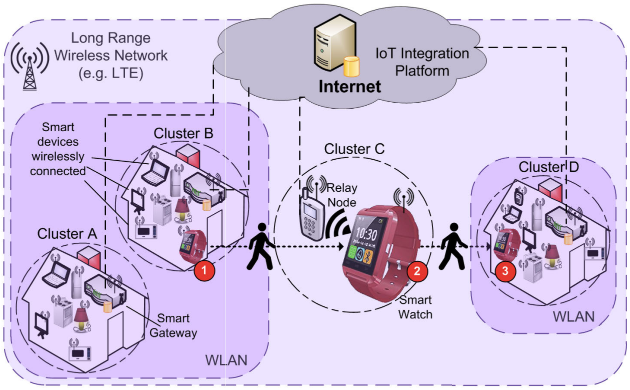


**FIGURE 1.** REMOS-IoT overview.

REMOS-IoT architecture. IoT objects are interconnected via the smart gateways. Smart gateways form device clusters in order to improve device intercommunication, identify low-performing and moving devices and manage admission control. ITINP manages smart gateways and relay nodes, and increases clusters performance by computing objects performance metrics then re-arranging low-performing and moving objects into suitable smart gateways or relay nodes.

The REMOS-IoT algorithms aim to improve the performance of communication by connecting devices to the best available smart gateways or to selected relay nodes. These algorithms consider diverse factors including distance between devices and smart gateways, and estimation of communication performance based on QoS metrics such as packet loss, delay and throughput. They also take into consideration device mobility, communication relevance and age of information. It is expected that by the use of REMOS-IoT, devices present a decrease in packet loss and delay and achieve higher throughput, while maintaining connectivity of moving devices.

Although other IoT architectures have also considered QoS, to the best of authors' knowledge, they have not employed parameters such as object mobility, relevance and age of information. The proposed REMOS-IoT architecture and solution integrate these and QoS metrics and use them in innovative algorithms in order to improve the overall IoT network performance.

The REMOS-IoT architecture and algorithms will be described in details in the following sections.

Extensive testing was performed using the Network Simulator 3 (NS-3) in the context of the scenario illustrated in Fig. 1. The scenario considers a mobile user with a smart watch moving from one wireless local area network (WLAN) (step 1 in Fig. 1) towards another WLAN (step 3). However, while moving, the device needs to use a relay node to remain connected (step 2 in Fig. 1), as it is out of WLAN cover. Testing results indicate how by employing REMOS-IoT, the QoS performance of IoT device connectivity increases.

This paper is organized as follows. Section II describes related works and section III introduces the architecture, components and algorithms of REMOS-IoT. Section IV presents the testing setup, performance analysis and results. Section V concludes the paper.

## II. RELATED WORK

In this section, works related to IoT QoS, clustering techniques, D2D architectures for IoT and mobility are discussed. These works build the foundation for the development of REMOS-IoT.

### A. QoS IN THE INTERNET OF THINGS

QoS limitations affect IoT, requiring solutions to optimize services and network delivery. In [5], authors proposed a solution for service-oriented IoT, consisting of a layered QoS scheduling model with sensing, network and application layers. Metrics used in the sensing networks include information precision, energy, sensing accuracy, network life-time and overhead costs. In addition, authors presented a framework for services, devices and network evaluation, with algorithms for decision-making in each layer. Testing demonstrated improved QoS levels and longer node life-time.

Authors in [15] tested a network protocol stack and experimental network setup, the LoRa FABIAN. QoS metrics such as Signal-to-Noise ratio (SNR), packet error rate (PER) and Received Signal Strength Indicator (RSSI) were evaluated in multiple scenarios. Results demonstrated that the location and elevation of antennas affect quality of network. Authors have yet to analyze combined traffic of down and uplink to determine a possible correlation when selecting optimal stations for nodes.

A congestion control scheme for IoT presented in [16] aims to achieve better delivery in applications with high packet loss. The solution, developed for the Constrained Application Protocol (CoAP), contains a back-off timer, traffic prioritization and adaptive re-transmission times.

Authors in [17] proposed a predictive algorithm for dynamic virtual machines (VM) allocation. Authors considered a cloud IoT scenario, analyzing QoS with a metascheduler architecture. QoS metrics such as costs and deadlines determine the allocation of VMs, in heterogeneous networks with low resource availability. Results indicated that the solution was able to comply with cost and time requirements. Authors, however, have yet to employ features such as other workload models, admission control and VMs migration.

A cost-effective analytical model was presented in [18]. Authors employed a finite capacity queuing system with preemptive resume service priority and push-out buffer management in the scheme. Application scenarios with low delay needs were considered, so the model should prioritize data sensitive to delays. Testing indicated increase in performance of priority traffic and the buffering technique avoided loss of packets containing emergency data.

### B. CLUSTERING IN THE INTERNET OF THINGS

Clustering in IoT is used to increase network performance and connect devices in a smarter and more efficient way. An IoT solution for lightweight virtualization was introduced in [19]. Applications were integrated in a distributed and virtualized environment. Authors compared two frameworks for IoT service provisioning: one based on direct interaction between two cooperating devices and one with a manager supervising the operations between cooperating devices forming a cluster. A testbed was implemented, and testing evaluated power and resource consumption. Tests indicated that lightweight virtualization allows for the execution of a wide range of IoT applications, while enabling the desired abstraction level with advantages in terms of manageability and scalability.

Authors presented in [20] a matching-value based IoT service composition scheme. The work focused on the representation, discovery, detection, and composition of services. Authors also introduced an algorithm for cluster-based distribution, allowing more robust and trusty decision process, with a distributed consensus method in order to improve robustness and trustiness of the decision process. The scheme, however, did not considered all phases of services life cycle.

An architecture for deploying wireless sensor networks (WSN), with several sink nodes and layers, was proposed in [21]. The architecture considers IoT-cloud integration and also included a routing protocol and an algorithm for selecting cluster heads in multi-layer multi-sink clusters. The algorithm performs load balancing while considering load fairness and energy. Results indicated improvements in network life time, even though the solution does not include peer-to-peer (P2P) links.

### C. D2D ARCHITECTURES FOR THE INTERNET OF THINGS

D2D communications are essential in order to provide support for IoT heterogeneous device interconnection. A modular architecture for IoT with D2D features was proposed in [12]. The architecture contains a direct D2D communication sub-layer. Authors use D2D to address the needs of the next generation IoT applications, such as low latency, robustness of connection, support of different data types, re-configurability and wide coverage.

A hybrid resource allocation algorithm for D2D communications was proposed in [13]. A centralized interference mitigation algorithm and a distributed joint channel selection and power allocation algorithm were proposed in order to achieve a near-zero infeasibility ratio of the spectrum capacity due to interference, while improving energy efficiency.

Authors in [14] introduced an overlay-based relay assisted D2D communications IoT architecture. The solution improves cellular networks, and by adopting low power pico base stations for relay, it facilitates D2D communications. Results showed that the utilization of green energy was maximized by balancing the energy among the relay base stations.

### D. MOBILITY IN THE INTERNET OF THINGS

Several IoT devices are mobile, which brings several challenges for researchers. A framework for path planning in emergencies was proposed in [22]. The framework considers mobile IoT devices and indoor human movement, in order to achieve optimal evacuation times. Results demonstrated that the prototype performs better than other existing solutions.

Authors in [23] introduced an agile data delivery framework for smart cities. The framework contains multimedia applications for vehicular networks (VANETs) and WSNs. The solution aims to select optimized paths for packets while maintaining high QoS levels. Metrics studied include energy, delay and throughput, and results indicated less overall delay and longer network life time. The effects of the framework in a larger network have yet to be studied.

Authors proposed a social IoT (SIoT) architecture for handoff between mobile and fixed access points in [24]. Android devices can share, download and use geo-information such as travel information of scenic spots, landmarks, and other points of interest during group tours. Tests indicated lower energy consumption, service time and cost.

REMOS-IoT was designed to push further the state-of-the-art represented by the approaches described in this section. It focuses on device mobility, D2D communications, object clustering and QoS-awareness. REMOS-IoT employs novel innovative algorithms which offer mobility support for diverse IoT service, while maintaining high levels of QoS.

## III. REMOS-IoT ARCHITECTURE

The REMOS-IoT architecture consists of clustered IoT devices, smart gateways, and the IoT Integration Platform (ITINP), a cloud-based server application. This architecture extends our previous work described in [25], [26] by providing mobility support, D2D communications and enhanced QoS awareness via the following metrics: age of information, location, device relevance and performance (*i.e.* a score combining loss, delay and throughput).

REMOS-IoT architectural components (*i.e.* IoT devices, smart gateways and ITINP) contribute to different layers in the five-layer architecture proposed in [27].

Fig. 2 illustrates the major REMOS-IoT components in the context of these five IoT architectural layers. The *Objects layer* contains the IoT objects, such as appliances and smart sensors, and supports basic D2D communication links between them. The *Network layer*, which contains the smart gateways, enables further communication support between objects, complementing D2D communications. The smart gateways cluster local IoT objects around them, and try to deploy solutions to improve overall communication performance by improving network conditions. The *Service Management layer* includes ITINP, which receives performance data from the smart gateways. As a proper Service Management layer component, ITINP performs smart control of network communications and involves management of the exchange of multiple data types such as multimedia content and sensor feedback. The *Applications layer* deploys user applications and allows for remote communications in the context of various user services. The *Business layer* contains Big Data applications, which are mainly used for

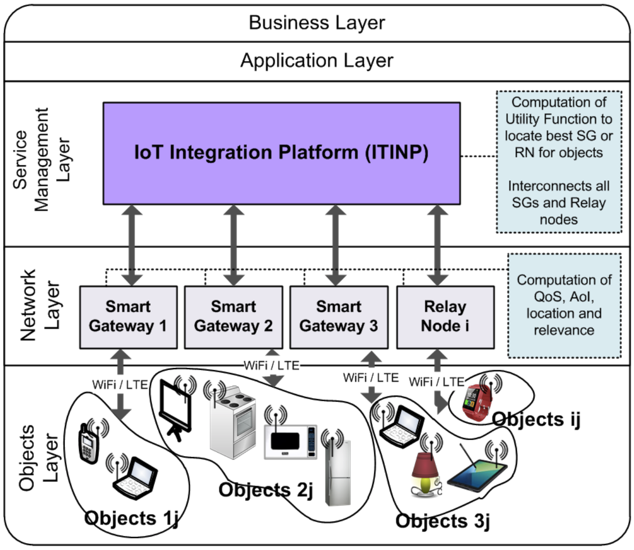


**FIGURE 2.** **REMOS-IoT layered architecture.**

policy-related decision-making. REMOS-IoT does not have components at upper two IoT architectural layers.

The following sub-sections present in details REMOS-IoT and its components, including utility functions, reputation score, and algorithms for mobility support and clustering.

### *A. REMOS-IoT MAIN ARCHITECTURAL COMPONENTS*

IoT devices such as wearables (*e.g.* smart watches, wrist bands), smartphones, sensors, robots, drones and smart appliances, offer a wide range of services for users and other devices. In a cluster of objects, certain objects can be more relevant to others, depending on the amount of service requests received within the cluster. Objects can avail themselves of long or short range connections (*e.g.* Long-Term Evolution (LTE) or WiFi) according to positioning, connection availability, performance and relevance of the object to a cluster, which can be physically near or far from the object. Objects connected to a smart gateway or relay node ($SG/RN_i$) receive an ID ($O_{ij}$) and are able to provide one or more services $S_{ijk}$, where $i$ indicates the gateway or relay node, $j$ identifies the object and $k$, the service. Objects are initially attached to the closest smart gateway, creating a cluster defined by distance. Object services requested often by devices from other clusters, might indicate that the object is more related to another cluster. In this scenario, the object might be using the current gateway solely as a bridge for communications, affecting performance. Therefore, it is important a regular reassignment of each object to the most relevant smart gateway. When the most relevant smart gateway is located outside of that range of the object, a long range wireless network (*e.g.* LTE) will be selected for smart gateway-object communications, creating a logical cluster. REMOS-IoT contains an algorithm for mobility support, balancing location and performance, while considering the availability of relay nodes (*e.g.* smartphones and other objects able to share internet access) for networking mobile IoT devices.

Smart gateways interconnect IoT objects, manage admission control, calculate and store performance-related information and are responsible for decision-making in relation to objects. IoT objects communicate to each other through the smart gateways that interconnect them. In order to allow devices to access services provided by objects connected to other smart gateways, the cloud-based ITINP is used, as it interconnects all smart gateways from REMOS-IoT network. Smart gateways are aware of devices' locations, supporting their mobility. Locations are used in the clustering algorithms and for admission control. As seen in Fig. 3, the object $O_{23}$ is moving in the direction of an area covered by two access points and a base station. In order to determine which access point or base station will reply to the connection request of the object, gateways calculate the score that elects the best gateway to reply first. For example, object $O_{21}$ is closer to $BS_3$, but according to the object's reputation and $BS_3$ available resources, the device is better connected to $AP_2$.

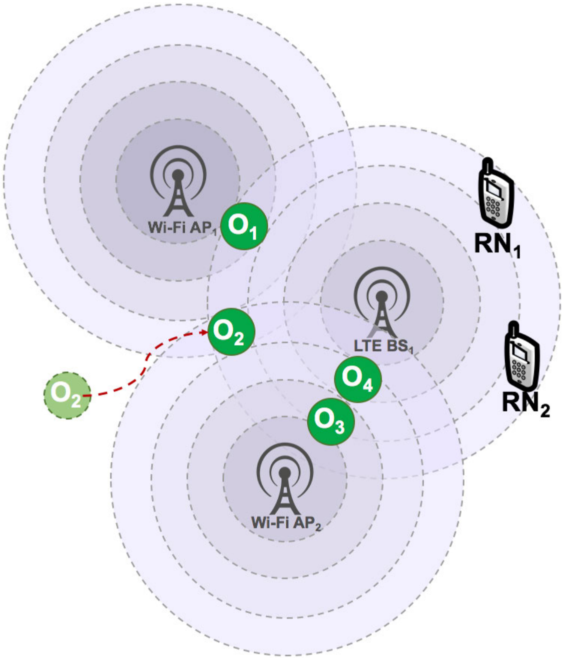


**FIGURE 3.** **Smart gateway admission control scheme.**

ITINP is a cloud-based server in charge of forming and managing improved clusters of IoT devices, connecting them to the best available smart gateway. ITINP achieves this by computing objects' performance, and by using an algorithm for re-clustering. ITINP has smart gateways' locations, therefore, if a device cannot reach the most suitable gateway using a short range wireless connection (*e.g.* WiFi), the device will either be requested to connect itself to the new gateway using a long range wireless technology (*e.g.* LTE), or be requested to attach itself to an available relay node, in order to remain connected to ITINP and therefore to the rest of the network.

### B. COMPONENTS' UNITS AND ALGORITHMS

Each component of REMOS-IoT architecture, *i.e.* IoT objects, smart gateways and ITINP, contains units responsible for gathering of QoS information, data processing, and decision making, as illustrated in Fig. 4.

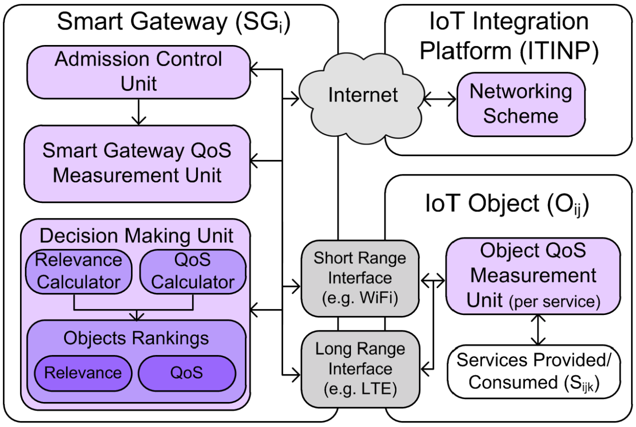


**FIGURE 4.** **REMOS-IoT major component units.**

Each IoT object's QoS Measurement Unit is responsible for collecting regularly QoS data from the services available in the device (in devices able to run the collection algorithm). This data is then sent to the smart gateway, which weights, normalizes and averages the values, calculating QoS device scores. The metrics considered in this paper are packet loss ratio, delay and throughput.

The Smart Gateway's Admission Control Unit assesses objects' reputations in order to accept or reject devices. It also checks the gateway performance (computed by Smart Gateways' QoS Measurement Unit) in order to determine whether it can support a new incoming device or not. If it cannot support extra devices, ITINP will identify the next most suitable smart gateway for the device. Locations of devices are also measured in this unit, and then informed to the Decision Making Unit.

The Smart Gateway's QoS Measurement Unit retrieves the QoS metrics of devices that do not have the resources to run the QoS data collection algorithm themselves. These metrics are collected per service. This unit is also responsible for collecting the QoS metrics from the smart gateways, so the gateway performance can be analyzed for admission control of incoming devices. This unit also computes the age of information in relation to the communications between the gateway and devices.

The Smart Gateway's Decision Making Unit is responsible for computing metrics and ranking devices in relation to relevance, QoS, location and Age of Information scores. It also identifies low-performing or moving objects that might be re-assigned to other gateways or relay nodes. Functions presented in eq. (1) and eq. (2), in section III.C, are evaluated in the QoS Calculator unit, whereas the Relevance Calculator computes the number of packets exchanged between devices and gateways, determining which clusters are closely related to a device. The computed scores are stored at the level of the smart gateway, which also stores their maximum values required for normalization.

The ITINP Networking Scheme deploys two innovative heuristic algorithms, one for object mobility support, and one for re-clustering low-performing objects.

#### 1) OBJECT MOBILITY SUPPORT ALGORITHM

The first algorithm, the *Object Mobility Support algorithm*, is a handover algorithm which extends the scheme introduced in [28] and is described in Algorithm 1. The algorithm performs mobility detection, allowing gateways and relay nodes to proactively disconnect and reconnect devices. The mobility support algorithm is triggered when the average RSSI or Reference Signal Received Power (RSRP) ($L_{ij}$) of devices reach a minimum threshold (set by default to -90 dBm) in relation to their current gateway. The gateway broadcasts a message with the object ID and its location score to neighboring gateways/relay nodes, and the object that is moving also receives this message. The goal is to receive messages from other gateways and relay nodes to select a good option for the device moving away. The object with low RSSI/RSRP, then, also sends a broadcast message, so nearby gateways and relay nodes can calculate their distances in relation to the object and verify the reputation score of the device in relation to them. These gateways, which also received a message from the device's current gateway, send a message containing the calculated location and reputation scores in relation to the object back to the current gateway, so the gateway can select the future gateway with the best combination of location and reputation. The current gateway notifies the target gateway, which attaches the object before the current gateway can detach it.

As novel aspects, the proposed Object Mobility Support algorithm uses new performance metrics based on device reputation and location as part of an utility function ($U_{ij}$) in the process of improved gateway selection. The utility

**Algorithm 1** Providing Mobility Support for Objects

```
if ($L_{ij} <= threshold$) then
    $SG_i$ broadcasts message with $O_{ij}$ and $L_{ij}$
    $SG_i$ waits other $SGs$ and $RNs$ (within $O_{ij}$ range) reply
    $O_{ij}$ receives message broadcasted by $SG_i$
    $O_{ij}$ broadcasts message to $SGs$ and $RNs$
    $SGs$ and $RNs$ compute $L_{ij}$ received from $O_{ij}$
    $SGs$ and $RNs$ send message with $L_{ij}$ and $U_{ij}$ to $SG_i$
    $SG_i$ selects best $L_{ij} + U_{ij}$
    if ($SG_i$ has best $L_{ij} + U_{ij}$) then
        return
    end
    else
        $SG_i$ notifies $SG/RN$ with best $L_{ij} + U_{ij}$
        $SG/RN$.attach($O_{ij}$)
        $SG_i$.detach($O_{ij}$)
    end
end
```

function ($U_{ij}$) is detailed in section III.C. It also makes use of D2D communications with available relay nodes in order to maintain connectivity of mobile IoT devices outside smart gateway range. This enables the possibility of finding either available well-performing gateways or relay nodes to keep device connection alive, when the current communication signal reaches a minimum threshold level.

#### 2) RE-CLUSTERING OF LOW-PERFORMING OBJECTS ALGORITHM

The second REMOS-IoT algorithm, described in Algorithm 2, is the *Re-Clustering of Low-Performing Objects algorithm*. This algorithm finds better gateways for low-performing objects and matches them with the most relevant gateway (*i.e.* the one with most devices consuming/providing services to the object). Devices not attached to their closest (according to the $L_{ij}$ metric) and most relevant ($R_{ij}$) gateway are reconnected to a suitable gateway. Performance ($P_{ij}$) and age of information ($A_{ij}$) are other metrics combined in the final reputation score ($U_{ij}$) used for improving clusters. These scores are detailed in section III.C. The algorithm expects to attach devices to their most relevant gateway, when devices are not yet attached to them. The performance of the most relevant gateway ($P_i$) is also considered, as devices will not be re-attached if the most relevant gateway has poor performance. In relation to devices performing poorly, based on their performance scores, the algorithm tries to attach these devices to better performing gateways, locating the next most relevant gateway if the current one is the most relevant.

**Algorithm 2** Re-Clustering Low-Performing Objects

```
Require: $P_{ij}$, $R_{ij}$, $A_{ij}$ and $L_{ij}$ per $O_{ij}$ in $SG_i$
foreach ($O_{ij}$) do
    ITINP computes $U_{ij}$
    if ($SG_i$ is not max $L_{ij}$) or ($SG_i$ is not most relevant SG and most relevant SG $P_i >= SG_i\ P_i$) then
        most relevant SG.attach($O_{ij}$)
        $SG_i$.detach($O_{ij}$)
    end
    if ($O_{ij}$ has low $U_{ij}$ and most relevant SG = $SG_i$) then
        next most relevant SG.attach($O_{ij}$)
        $SG_i$.detach($O_{ij}$)
    end
end
```

Heuristic approaches were employed in the two algorithms, instead of analytical solutions due to the nature of the IoT device selection problem, which is NP-hard, as demonstrated in other works focused on IoT performance optimization [29]–[32].

### C. REPUTATION SCORE AND UTILITY FUNCTIONS

One of REMOS-IoT key features is the reputation score ($U_{ij}$). The score $U_{ij}$ combines QoS, relevance, age of information and location scores of REMOS-IoT. The reputation score is employed in admission and performance control (which is achieved by improving device clusters connecting them to suitable gateways or relay nodes) and detection of low-performing objects. This score allows REMOS-IoT algorithms to create improved device clusters as it indicates which devices are performing poorly and are affecting the overall network performance. Regarding device mobility, current and neighboring gateways and relay nodes exchange control messages and based on device reputation and location scores, the best gateway or relay node is selected to support moving devices communication.

Every service provided by a device has a performance score ($P_{ijk}$) as proposed in [25] and demonstrated in eq. (1). $P_{ijk}$ is a normalization of the sum of the normalized values of each metric $M_x$ measured per service (a device can provide one or more services). These performance scores are averaged to compose the device QoS score ($P_{ij}$), as seen in eq. (2). Smart gateways maintain the scores of current and previously connected devices. The QoS score of a smart gateway ($P_i$) is also calculated using this formula.

$$P_{ijk} = \sum_{x} \left( \frac{{M^{ijk}}_x}{\max\limits_{x}({M^{ijk}}_x)} \right) \tag{1}$$

$$P_{ij} = \underset{k}{avg}(P_{ijk}) \tag{2}$$

The Relevance score ($R_{ij}$) of a device, proposed in [25], is an array of tuples formed by the ID of each smart gateway that has interacted with the device and the percentage of total packet communication exchanged between the gateway and the device. For example, an object has sent and received 200 packets from objects located in the same smart gateway (*e.g.* $SG_1$) and 800 packets from objects located in a different one (*e.g.* $SG_2$). Therefore, the device relevance score is $R_{ij}$=[$SG_1$,20%],[$SG_2$,80%].

The Age of Information score ($A_{ij}$) is a performance metric that indicates the freshness of information, and it is also included in the scores which compose the reputation score. The average age of information of a device is determined by a function for the M/M/1 queue model, given in [33] as seen in eq. (3). The arrival rate (which indicates the generation and submission of packets) is determined by a Poisson process $\lambda$ and average service time is given by $1/\mu$. The utilization $\rho = \lambda/\mu$ varies according to each service provided by the devices.

$$A_{ij} = \frac{1}{\mu}\left(1 + \frac{1}{\rho} + \frac{\rho^2}{1-\rho}\right) \tag{3}$$

The Location score ($L_{ij}$) of a device is based on its RSSI or RSRP in relation to the neighboring gateways or relay nodes (in D2D communications).

REMOS-IoT uses the RSSI collected by gateways in Wi-Fi access points (AP) and RSRP collected by gateways in LTE base stations (BS) as the metrics for device location. These metrics were chosen because even simple IoT devices such as beacons, provide RSSI. For distance estimation, RSSI-based methods are useful due to their low cost, low power and accessibility. They are already used in many systems.

REMOS-IoT considers -90 decibel-milliwatts (dBm) to be a low RSSI, taking decisions when devices present low dBm, which means that they are likely moving away from the gateway.

The overall utility function used in the calculation of the reputation score is presented in eq. (4). REMOS-IoT computes the reputation of a device in relation to the smart gateways in order to find a balance between performance, age of information, relevance and location of devices, resulting in a normalized $U_{ij}$ reputation score per object. The sum of weights equals 1.

$$U_{ij} = wp\frac{P_{ij}}{\max\limits_{j}(P_{ij})} + wr\frac{R_{ij}}{\max\limits_{j}(R_{ij})} + wa\frac{A_{ij}}{\max\limits_{j}(A_{ij})} + wl\frac{|L_{ij}|}{\max\limits_{j}(|L_{ij}|)} \quad (4)$$

The location score $L_{ij}$ is used to support device mobility (see Algorithm 1). The $U_{ij}$ score reveals how devices are performing and enables to identify those which perform badly affecting cluster performance, as seen in the re-cluster algorithm (see Algorithm 2).

## IV. PERFORMANCE ANALYSIS

For testing purposes, REMOS-IoT was modelled and evaluated using the NS-3 simulator [34], as illustrated in Fig. 5. The parameters used in the simulation setup including simulation duration, mobility speed, data rates, and mobility and antenna models and their values are presented in Table 1. Equal weights were applied to the reputation score ($U_{ij}$) used in REMOS-IoT algorithms (*i.e.* $wp = wr = wa = wl$).

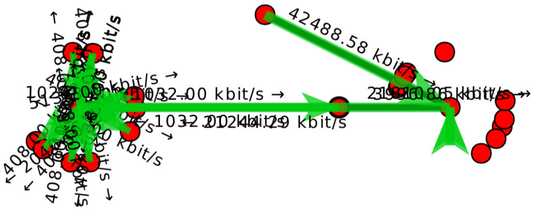


**FIGURE 5.** The NS-3 visualization mode.

**TABLE 1.** Simulation setup.

| Parameter | Value |
|---|---|
| Simulator | NS-3.24.1 |
| Duration of the Simulation | 14s+10s before and after sim. |
| Initial dist. between nodes and antennas | 3 metres |
| WiFi and LTE Data Rates | 40 Mbps and 100 Mbps |
| WiFi Standard | 802.11ac (40MHz, MCS 9) |
| LTE eNB Antenna Model Type | Isotropic Antenna Model |
| Remote Station Manager | ConstantRateWifiManager |
| Mobility Model | ConstantVelocityMobilityModel |
| Speed of Smart Watch user | 2 metres per second |

**TABLE 2.** REMOS-IoT scenarios description.

| **11 devices in WiFi + 11 devices in LTE** | | | | | |
|---|---|---|---|---|---|
| No. Devs. | Device Type | Scenario1 (Mbps) | Scenario2 (Mbps) | Scenario3 (Mbps) | Traffic Direction |
| 1 | High Bitrate | 10 | 20 | 30 | DL only |
| 1 | Avg. Bitrate | 5 | 10 | 15 | DL only |
| 1 | Low Bitrate | 2 | 4 | 6 | DL only |
| 7 | Very Low | 0.4 | 0.8 | 1.2 | UL & DL |
| 1 | Mobile Device | 1 | 2 | 3 | UL & DL |
| Total Bitrate | | 20.8 | 41.6 | 62.4 | |
| **22 devices in WiFi + 22 devices in LTE** | | | | | |
| No. Devs. | Device Type | Scenario4 (Mbps) | Scenario5 (Mbps) | Scenario6 (Mbps) | Traffic Direction |
| 2 | High Bitrate | 5 | 10 | 15 | DL only |
| 2 | Avg. Bitrate | 2.5 | 5 | 7.5 | DL only |
| 2 | Low Bitrate | 1 | 2 | 3 | DL only |
| 14 | Very Low | 0.2 | 0.4 | 0.6 | UL & DL |
| 2 | Mobile Device | 0.5 | 1 | 1.5 | UL & DL |
| Total Bitrate | | 20.8 | 41.6 | 62.4 | |
| **100 devices in WiFi + 100 devices in LTE** | | | | | |
| No. Devs. | Device Type | Scenario7 (Mbps) | Scenario8 (Mbps) | Scenario9 (Mbps) | Traffic Direction |
| 95 | Very Low | 0.208 | 0.416 | 0.624 | UL & DL |
| 5 | Mobile Device | 0.208 | 0.416 | 0.624 | UL & DL |
| Total Bitrate | | 20.8 | 41.6 | 62.4 | |

Nine scenarios, as demonstrated in Table 2, were designed with clusters representing smart houses/businesses containing IoT objects. Mobile devices (exemplified as smart watches) are located at a smart home with other IoT objects all interconnected by a local gateway using a WiFi 802.11ac access point. These mobile devices move towards another smart home and when out of the reach of both gateways, it needs to connect to a relay node using D2D communications to continue online.

Each gateway of the two smart homes, one connected to WiFi access points and one connected to an LTE base station, support a number of devices types in WiFi and LTE networks: three smart objects of high, average and low bitrates (*e.g.* in scenario 2, these device types have 20, 10 and 4 Mbps, respectively, representing video devices) with downlink only; and seven smart objects very low rates, simulating diverse mobile and non-mobile IoT objects. The mobile devices in the scenarios move from the WiFi LAN to the LTE network. When the device is in between networks, out of their covered area, an available relay node provides a D2D shared connection, so the smart watch is still under the ITINP platform.

Besides the multiple device types in each scenario, different amount of devices were tested: 22 devices in scenarios 1, 2 and 3; 44 devices in scenarios 4, 5 and 6; and 200 devices in scenarios 7, 8 and 9. Half of devices were located in the WiFi network and the other half in the LTE network. The mobile devices start in the WiFi network, moving towards the LTE network. In the scenarios with 200 devices (*i.e.* 7, 8 and 9) only device types of very low bitrate were included, representing an IoT network with numerous constrained devices.

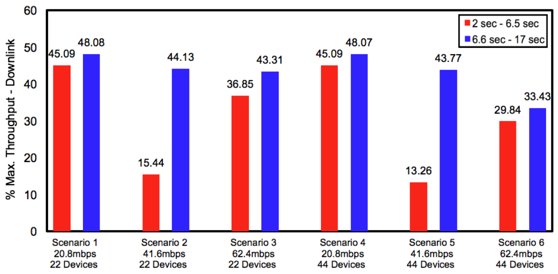


**FIGURE 6.** WiFi - Downlink - % Max. Thru. of devices with high consumption.

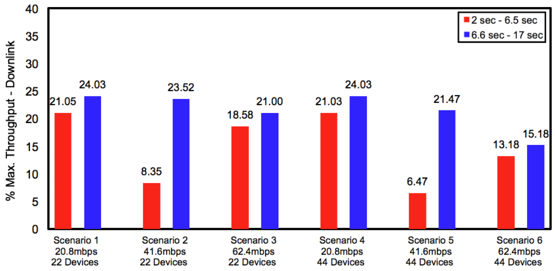


**FIGURE 7.** WiFi - Downlink - % Max. Thru. of devices with avg. consumption.

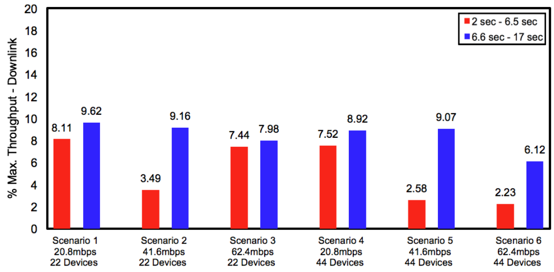


**FIGURE 8.** WiFi - Downlink - % Max. Thru. of devices with low consumption.

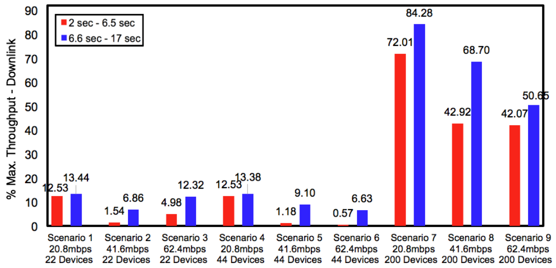


**FIGURE 9.** WiFi - Downlink - % Max. Thru. of Devs. with very low cons.

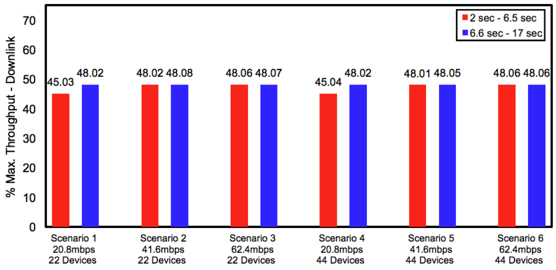


**FIGURE 10.** LTE - Downlink - % Max. Thru. of Devices with High Consumption.

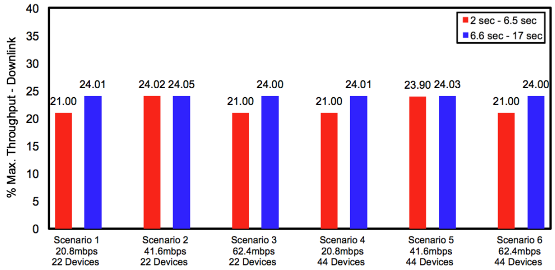


**FIGURE 11.** LTE - Downlink - % Max. Thru. of Devices with Avg. Consumption.

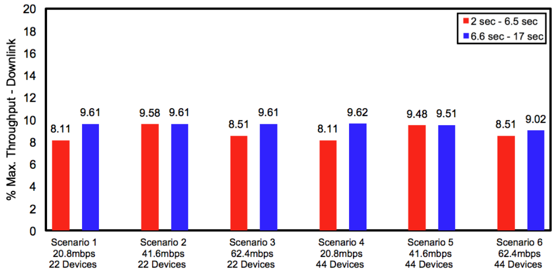


**FIGURE 12.** LTE - Downlink - % Max. Thru. of devices with low consumption.

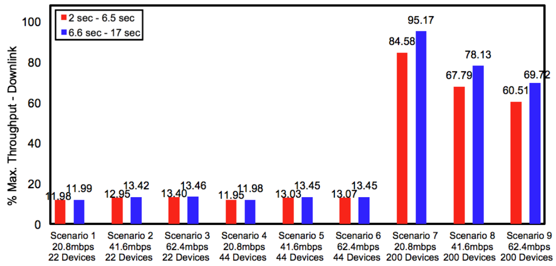


**FIGURE 13.** LTE - Downlink - % Max. Thru. of Devs. with very low cons.

Three levels of total bitrate were shared among the different device types in each network. Scenarios 1, 4 and 7 have a low load of 20.8 Mbps when all bitrates are summed; scenarios 2, 5 and 8 have an average load 41.6 Mbps available for the devices; and scenarios 3, 6 and 9 have a high load of 62.4 Mbps divided among devices. The scenarios also include the other REMOS-IoT features, such as regular analyses of performance and objects' locations, re-clustering devices affecting network performance in order to increase quality of data delivery for the entire IoT network under ITINP's domain.

### *A. SCENARIOS OVERALL RESULTS*

The plots from Figs. 6 – 29 present comparisons between the nine scenarios. The red bars show results from the seconds 2 until 6.5, before mobile devices moved into the LTE

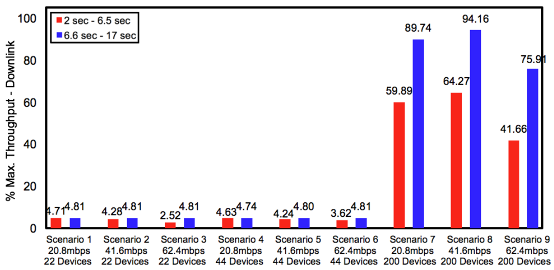


**FIGURE 14.** **Downlink - % Max. Thru. of devices moving from WiFi to LTE.**

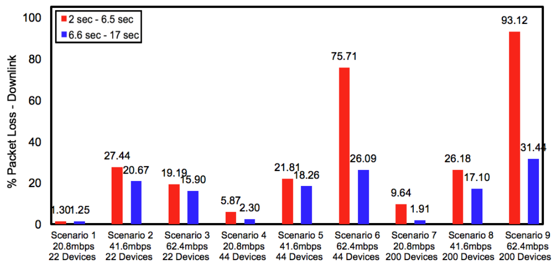


**FIGURE 15.** **WiFi - Downlink - % Avg. Pkt. loss of fixed devices.**

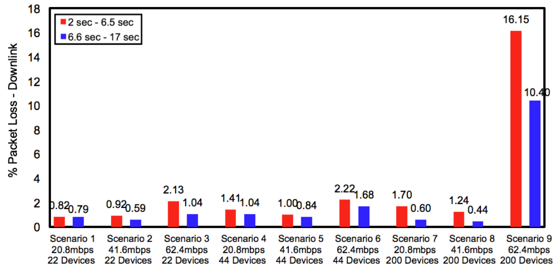


**FIGURE 16.** **LTE - Downlink - % Avg. Pkt. loss of fixed devices.**

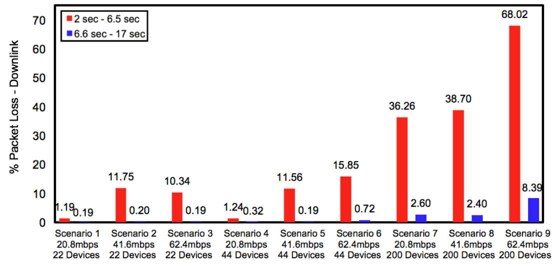


**FIGURE 17.** **Downlink - % Avg. Pkt. loss of devices moving from WiFi to LTE.**

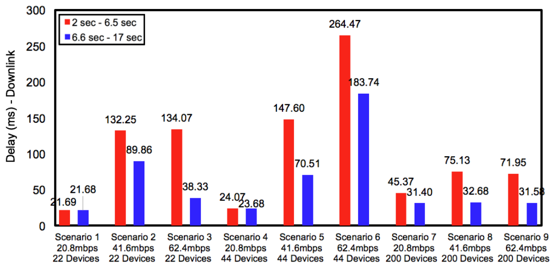


**FIGURE 18.** **WiFi - Downlink - Avg. Delay (ms) of fixed devices.**

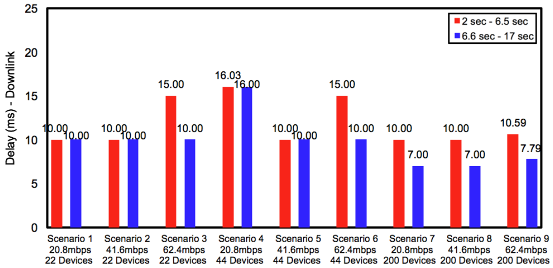


**FIGURE 19.** **LTE - Downlink - Avg. Delay (ms) of fixed devices.**

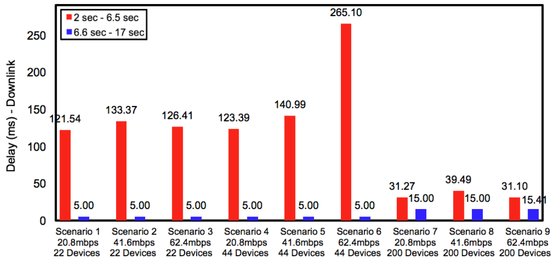


**FIGURE 20.** **Downlink - Avg. Delay (ms) of devices moving from WiFi to LTE.**

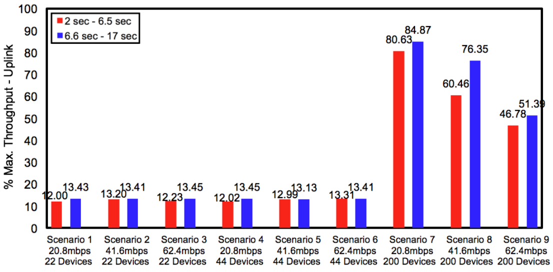


**FIGURE 21.** **WiFi - Uplink - % Max. Thru. of Devs. with very low consumption.**

network. The blue bars represent seconds 6.6 until the end of simulation in second 17, after mobile devices moved and REMOS-IoT algorithms were performed for relay and mobility support, and analysis was performed for downlink and uplink traffic.

With a continued connection for mobile devices, there is a seamless change from one network to another. High throughputs being achieved in most scenarios, in both uplink and downlink. Scenario 9, which contains a high number of devices and high bitrate requirements (*i.e.* 200 devices and 62.4 Mbps total bitrate), had less benefits in relation to throughput than other scenarios, due to the amount of congestion. Still, mobile devices, for instance, had an increase of 83% of achieved throughput in this scenario.

Delays in both uplink and downlink scenarios in WiFi and LTE were greatly decreased. In scenario 6, for instance, a high delay of 265.1ms was decreased to 5ms for moving devices. Devices in WiFi, LTE and mobile ones had an average decrease of 43%, 17% and 92%, respectively, in delay

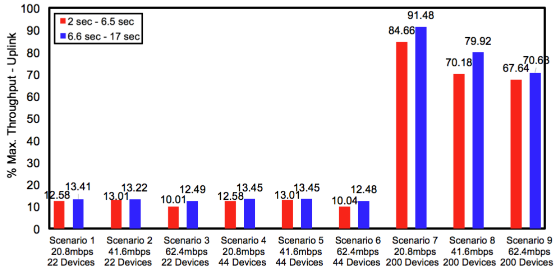


**FIGURE 22.** LTE - Uplink - % Max. Thru. of devices with very low consumption.

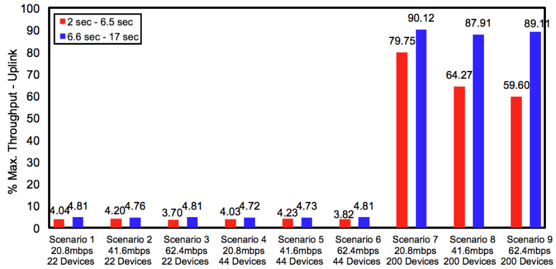


**FIGURE 23.** Uplink - % Max. Thru. of devices moving from WiFi to LTE.

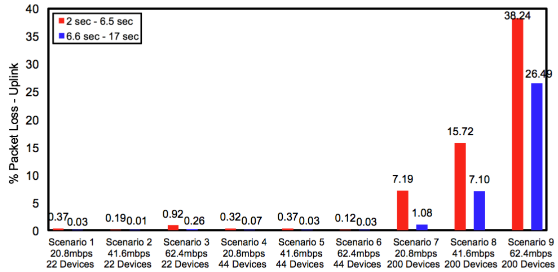


**FIGURE 24.** WiFi - Uplink - % Avg. Pkt. loss of fixed devices.

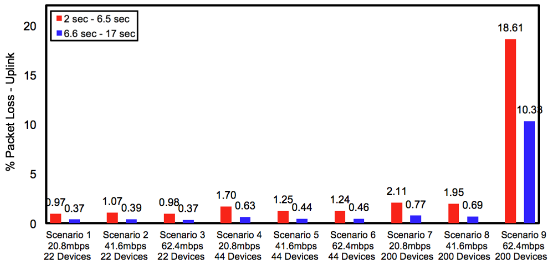


**FIGURE 25.** LTE - Uplink - % Avg. Pkt. loss of fixed devices.

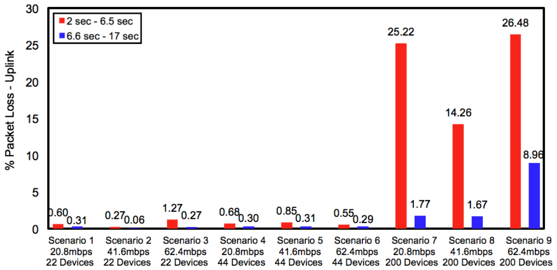


**FIGURE 26.** Uplink - % Avg. Pkt. loss of devices moving from WiFi to LTE.

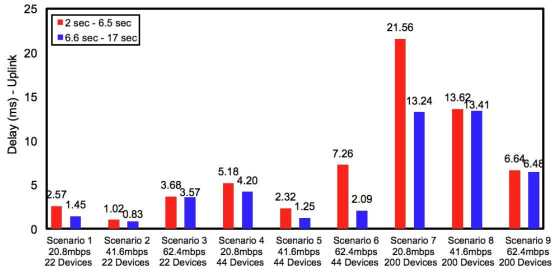


**FIGURE 27.** WiFi - Uplink - Avg. Delay (ms) of fixed devices.

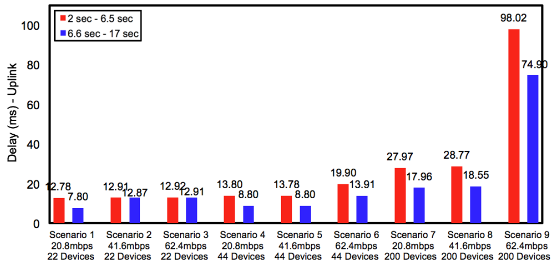


**FIGURE 28.** LTE - Uplink - Avg. Delay (ms) of Fixed Devices.

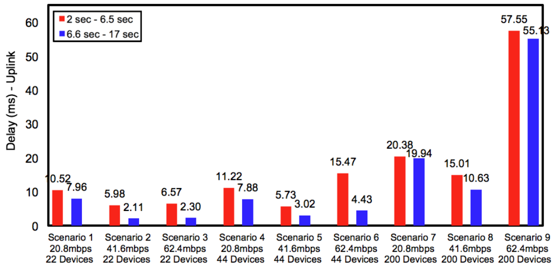


**FIGURE 29.** Uplink - Avg. Delay (ms) of devices moving from WiFi to LTE.

across all scenarios for downlink. For uplink, the decreases in delay were 27%, 26%, and 23% for WiFi, LTE and mobile devices.

Packet loss was also greatly reduced across all scenarios. On average in downlink, packet loss was decreased in 51%, 36% and 92% for WiFi, LTE and mobile devices, when considering all scenarios. In relation to uplink, loss was decreased in 44%, 51%, and 80% in WiFi, LTE and mobile devices.

The obtained results from the nine scenarios show that REMOS-IoT is able to perform significant improvements in terms of throughput, packet loss and delay, the selected metrics for the study.

### *B. BASELINE COMPARISON*

REMOS-IoT is compared against the IoT-RTP and IoT-RTCP adaptive protocols for multimedia transmission in IoT environments [35], in order to verify its gains in comparison with a recent IoT networking approach. IoT-RTP and IoT-RTCP are adaptive versions of the real-time transport protocol (RTP) and real-time control protocol (RTCP).

The IoT-RTP and IoT-RTCP protocols, deployed on the Network Simulator 2 (NS-2), employ a novel mechanism of dividing large multimedia sessions into simpler sessions with awareness of network status. Although REMOS-IoT was deployed on NS-3, significant differences were not found when comparing the schemes, in relation to the simulator used, with results being alike. For the baseline comparison, delays and packet losses were averaged over time in all devices in the second WLAN, after REMOS-IoT algorithms were applied. A similar simulation runtime was considered. The obtained results are available in Table 3. REMOS-IoT outperforms the other schemes with 96.3% lower delay and 82.6% less packet loss, considering the average load scenario 2 of REMOS-IoT. Statistical significance in favor of REMOS-IoT, in both delay and loss samples, is evidenced by the low p-value of $6.9 \times 10^{-35}$ for delay and $3.25 \times 10^{-6}$ for packet loss, obtained following a paired student t-test result analysis.

**TABLE 3.** **Baseline comparison.**

| | **Delay (ms)** | | **Loss (%)** | |
|---|---|---|---|---|
| | **IoT - RTP & IoT - RTCP** | **REMOS-IOT** | **IoT - RTP & IoT - RTCP** | **REMOS-IOT** |
| **Average** | 204.74 | 7.53 | 2.50 | 0.43 |
| **St. Dev.** | 2.83 | 0.11 | 1.34 | 0.08 |
| **Max. Value** | 211.00 | 7.75 | 5.00 | 0.60 |
| **Min. Value** | 200.00 | 7.25 | 1.00 | 0.35 |
| **T-test p-value** | $6.9 \times 10^{-35}$ (<0.05) | | $3.25 \times 10^{-6}$ (<0.05) | |

## V. CONCLUSION AND FUTURE WORK

This paper introduced REMOS-IoT, which enhances the existing IoT architecture with algorithms to improve the performance of mobile IoT device connectivity. REMOS-IoT records scores for QoS, relevance, age of information and location, in order to cluster IoT devices efficiently and increase their communication performance, while supporting device mobility. The testing scenario consists of a WiFi-only smart watch in a smart house with several other smart devices initially connected to a local gateway using a WiFi 802.11ac access point. The smart watch loses connection and needs to avail from the proposed REMOS-IoT algorithms in order to maintain high quality of service levels. The solution was tested via NS-3 modelling and simulations and results for uplink and downlink were recorded in terms of throughput, packet loss and delay. REMOS-IoT outperforms another solution in terms of these QoS metrics.

Future work includes an optimization of the relevance score by prioritizing different types of services and content.

## REFERENCES

[1] Ericsson. (2020). *Internet of Things forecast—Ericsson Mobility Report*. [Online]. Available: https://www.ericsson.com/en/mobility-report/reports

[2] N. Narendra and P. Misra. (2016). *Research Challenges in the Internet of Mobile Things*. [Online]. Available: https://iot.ieee.org/newsletter/march-2016/research-challenges-in-the-internet-of-mobile-things.html

[3] G.-M. Muntean, P. Perry, and L. Murphy, "Objective and subjective evaluation of QOAS video streaming over broadband networks," *IEEE Trans. Netw. Service Manage.*, vol. 2, no. 1, pp. 19–28, Nov. 2005.

[4] C. H. Muntean and J. McManis, "End-user quality of experience oriented adaptive E-learning system," *J. Digit. Inf., Special Issue Adapt. Hypermedia*, vol. 7, no. 1, pp. 1–13, 2006.

[5] L. Li, S. Li, and S. Zhao, "QoS-aware scheduling of services-oriented Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1497–1507, May 2014.

[6] S. Ezdiani, I. S. Acharyya, S. Sivakumar, and A. Al-Anbuky, "Wireless sensor network softwarization: Towards WSN adaptive QoS," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1517–1527, Oct. 2017.

[7] I. Yaqoob, E. Ahmed, I. A. T. Hashem, A. I. A. Ahmed, A. Gani, M. Imran, and M. Guizani, "Internet of Things architecture: Recent advances, taxonomy, requirements, and open challenges," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 10–16, Jun. 2017.

[8] E. Ahmed, I. Yaqoob, A. Gani, M. Imran, and M. Guizani, "Internet-of-Things-based smart environments: State of the art, taxonomy, and open research challenges," *IEEE Wireless Commun.*, vol. 23, no. 5, pp. 10–16, Oct. 2016.

[9] Y. Li, K. Chi, H. Chen, Z. Wang, and Y. Zhu, "Narrowband Internet of Things systems with opportunistic D2D communication," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1474–1484, Jun. 2018.

[10] Z. Zhou, K. Ota, M. Dong, and C. Xu, "Energy-efficient matching for resource allocation in D2D enabled cellular networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 5256–5268, Jun. 2017.

[11] Z. Zhou, H. Yu, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Dependable content distribution in D2D-based cooperative vehicular networks: A big data-integrated coalition game approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 953–964, Mar. 2018.

[12] H. Rahimi, A. Zibaeenejad, and A. A. Safavi, "A novel IoT architecture based on 5G-IoT and next generation technologies," in *Proc. IEEE 9th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON)*, Nov. 2018, pp. 81–88.

[13] Z. Zhou, M. Dong, K. Ota, G. Wang, and L. T. Yang, "Energy-efficient resource allocation for D2D communications underlaying cloud-RAN-based LTE—A networks," *IEEE Internet Things J.*, vol. 3, no. 3, pp. 428–438, Jun. 2016.

[14] X. Liu and N. Ansari, "Green relay assisted D2D communications with dual batteries in heterogeneous cellular networks for IoT," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1707–1715, Oct. 2017.

[15] T. Petric, M. Goessens, L. Nuaymi, L. Toutain, and A. Pelov, "Measurements, performance and analysis of LoRa FABIAN, a real-world implementation of LPWAN," in *Proc. IEEE 27th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2016, pp. 1–7.

[16] R. Hassan, A. M. Jubair, K. Azmi, and A. Bakar, "Adaptive congestion control mechanism in CoAP application protocol for Internet of Things (IoT)," in *Proc. Int. Conf. Signal Process. Commun. (ICSC)*, Dec. 2016, pp. 121–125.

[17] M. L. M. Peixoto, D. L. Filho, C. Henrique, D. Segura, B. Tardiole, and B. Guazzelli, "Predictive dynamic algorithm: An approach toward QoS-aware service for IoT-cloud environment," in *Proc. IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, Dec. 2016, pp. 686–693.

[18] I. Awan, M. Younas, and W. Naveed, "Modelling QoS in IoT applications," in *Proc. 17th Int. Conf. Netw.-Based Inf. Syst.*, Sep. 2014, pp. 99–105.

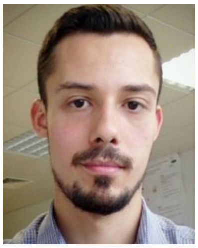
[19] R. Morabito, I. Farris, A. Iera, and T. Taleb, "Evaluating performance of containerized IoT services for clustered devices at the network edge," *IEEE Internet Things J.*, vol. 4, no. 4, pp. 1019–1030, Aug. 2017.

[20] S. Li, G. Oikonomou, T. Tryfonas, T. M. Chen, and L. Da Xu, "A distributed consensus algorithm for decision making in service-oriented Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1461–1468, May 2014.

[21] W. Twayej, M. Khan, and H. S. Al-Raweshidy, "Network performance evaluation of M2M with self organizing cluster head to sink mapping," *IEEE Sensors J.*, vol. 17, no. 15, pp. 4962–4974, Aug. 2017.

[22] L.-W. Chen and J.-J. Chung, "Mobility-aware and congestion-relieved dedicated path planning for group-based emergency guiding based on Internet of Things technologies," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2453–2466, Sep. 2017.

[23] F. Al-Turjman, "Energy-aware data delivery framework for safety-oriented mobile IoT," *IEEE Sensors J.*, vol. 18, no. 1, pp. 470–478, Jan. 2018.
[24] C.-M. Huang, C.-H. Shao, S.-Z. Xu, and H. Zhou, "The social Internet of Thing (S-IOT)-based mobile group handoff architecture and schemes for proximity service," *IEEE Trans. Emerg. Topics Comput.*, vol. 5, no. 3, pp. 425–437, Jul. 2017.
[25] A. A. Simiscuka, C. H. Muntean, and G.-M. Muntean, "A NETworking scheme for an Internet of Things integration platform," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2017, pp. 271–276.
[26] A. A. Simiscuka and G.-M. Muntean, "A relay and mobility scheme for QoS improvement in IoT communications," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2018, pp. 1–6.
[27] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.
[28] M. Bouaziz, A. Rachedi, and A. Belghith, "EC-MRPL: An energy-efficient and mobility support routing protocol for Internet of mobile things," in *Proc. 14th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2017, pp. 19–24.
[29] C.-C. Lin, D.-J. Deng, and A. L.-Y. Lu, "Many-objective sensor selection in IoT systems," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 40–47, Jun. 2017.
[30] A. Brogi and S. Forti, "QoS-aware deployment of IoT applications through the fog," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1185–1192, Oct. 2017.
[31] Q. Fan and N. Ansari, "Application aware workload allocation for edge computing-based IoT," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 2146–2153, Jun. 2018.
[32] F. Bian, D. Kempe, and R. Govindan, "Utility-based sensor selection," in *Proc. 5th Int. Conf. Inf. Process. Sensor Netw.*, 2006, pp. 11–18.
[33] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 2731–2735.
[34] 2018. *The NS-3 Network Simulator*. [Online]. Available: https://www.nsnam.org/
[35] O. Said, Y. Albagory, M. Nofal, and F. Al Raddady, "IoT-RTP and IoT-RTCP: Adaptive protocols for multimedia transmission over Internet of Things environments," *IEEE Access*, vol. 5, pp. 16757–16773, 2017.



**ANDERSON AUGUSTO SIMISCUKA** (Member, IEEE) received the B.Sc. degree in information systems from Mackenzie Presbyterian University, São Paulo, Brazil, in 2014, and the Ph.D. degree from the School of Electronic Engineering, Dublin City University (DCU), Ireland, in 2020. He is currently a Postdoctoral Researcher with the Performance Engineering Laboratory and the Insight SFI Centre for Data Analytics, School of Electronic Engineering, Dublin City University. He has worked in several telecom and software development projects in companies, such as Wittel, from 2010 to 2013, DCU/Ericsson (E-Stream Project), in 2014, Arkadin, in 2014, and IBM, in 2015. He is involved in the EU Horizon 2020-funded project TRACTION, which focuses on innovative technologies for Opera co-creation. His research interests include the Internet of Things (IoT) communications performance and rich media integration, such as VR, with the IoT devices. He is a member of the IEEE Young Professionals, the IEEE Communications Society, and the IEEE Broadcast Technology Society.

**GABRIEL-MIRO MUNTEAN** (Senior Member, IEEE) is currently a Professor with the School of Electronic Engineering, Dublin City University (DCU), Ireland, and a Co-Director of the DCU Performance Engineering Laboratory. He was the Project Coordinator of the EU-funded project NEWTON. He is the DCU Coordinator of the EU-funded project TRACTION. He has published more than 400 articles in top-level international journals and conference papers. He has authored four books and 23 book chapters, and edited nine additional books. His research interests include quality, performance, and energy saving issues related to multimedia and multiple sensorial media delivery, technology-enhanced learning, and other data communications over heterogeneous networks. He is an Associate Editor of the IEEE Transactions on Broadcasting, the Multimedia Communications Area Editor of the IEEE Communications Surveys and Tutorials, and a reviewer of important international journals, conferences, and funding agencies.

• • •