Analyzing Latency and Dropping in Today's Internet of Multimedia Things

Maha AlAslani and Basem Shihada
Computer, Electrical, and Mathematical Sciences & Engineering Division
KAUST, Saudi Arabia, {maha.aslani, basem.shihada}@kaust.edu.sa

***Abstract*—Internet of Multimedia Things (IoMT) applications such as real-time multimedia based security and monitoring in smart homes, hospitals, cities, and smart transportation management systems are of the most difficult systems to deploy. These services are highly time sensitive and require Quality of service (QoS) guarantees. QoS requirements are key factors that lead to variations of multimedia traffic quality and the Quality of Experience (QoE) for the end users. IoMT devices transmit measurements to a predefined IoMT application server subject to maximum QoS constraint. The delay and dropping are essential constraints as delayed packets are considered useless for the IoMT applications. Our objective is to obtain an approximate expression of the blocking probability due to either buffer overflow or violating certain end-to-end threshold. For this purpose, we employ M/G/1 framework for our network. We validate the proposed analytical model and demonstrate the blocking probability and end-to-end delay. We anticipate that our results are critical for optimizing IoMT network design and deployment.**

***Index Terms*—Internet of Things (IoT), Internet of Multimedia Things (IoMT), real time, multimedia traffic, queueing, end-to-end delay, dropping, quality of service (QoS), Quality of Experience (QoE)**

I. Introduction

The internet of things (IoT) is growing substantially. Connecting the physical reality with the virtual is opening new doors for innovation in almost every life discipline. The smart heterogeneous multimedia devices that interact and cooperate with one another and with other devices through the Internet create a novel paradigm called the Internet of Multimedia Things (IoMT). The quality of service (QoS) requirements of IoMT applications vary greatly, and the underlying network must be aware of these dynamic variations. From another hand, QoS focuses on the objective measurement of network parameters such as jitter, throughput, loss and delay only, without gives full attention on the user's experience of the service delivery quality (e.g., video bitrate, frame rate, the resolution of the mobile device, etc.). QoS is considered as an influencing factor for the Quality of Experience (QoE) perceived by the end users (machines or humans). For the multimedia traffic, the QoE adding a new dimension to provide a complete understanding of the system characteristics with respect to the network measures of the QoS. However, with the new advances in Software-Defined-Networking (SDN) and Information-Centric-Networking (ICN), network policies and applications deployment can be implemented on the spot and the dynamic changes on the network QoS policies and users QoE requirements can be reflected. And the network should make the correct decisions such as, which priority queue to use, which routing path to select, where to deploy the servers middleboxes and, etc. that would enable these packets to meet their requirements.

In this work, we provide the required analysis of an underlying network to serve real-time IoMT applications. Such analysis will show if the underlying network under various parameters can successfully serve the target IoMT application QoS and the users QoE requirements, if not, which parameters or knobs need to be tuned.

II. Related Works

With the advancement of today's next generation networks (NGN), the conceptual framework of providing various real-time services over the Internet (i.e., over the Internet service provider (ISP) networks) is becoming a reality. IoMT Systems represent one of the newly attractive applications for the IoT. IoMT services are real-time in nature and need to provide guaranteed quality and performance [1]. Such IoMT applications generate multimedia traffic that can be processed by remote servers (the cloud) and the delay caused by transferring data to the cloud and back to the application is unacceptable.

In the literature, few numbers of studies consider the concept of IoMT and their QoS, QoE requirements. Recently, fog computing has been introduced as a new technique of bringing the benefits of the cloud closer to where data is created which reduces the transmission delay [2] [3] [4]. To deploy the services efficiently at the edge of the networks, an analysis study needs to be presented. This paper aims at providing end to end analysis and calculating the required number of network hops to provide a guaranteed level of QoS for IoMT applications.

III. System Model

In this work, we consider a typical IoMT application scenario, in which M IoMT devices collect certain information, and then, transmit them over multiple switches to an IoMT application server. IoMT information i.e., data packet arrival is modeled as an exponential inter-arrival time with rates $\gamma_m, m = 1, 2, ..., M$. The intermediate switches are modeled as single server facilities with M/G/1 queue and two priority queues that are responsible for forwarding the IoMT data to the application server [5]. These switches are considered to be always running.

We categorize the IoMT packets into critical and non-critical packets and assume that the network is aware of the critical IoMT packets (through packet classification or flow matching), and thus, will schedule them to the higher priority queue denoted as Q_k^H, where $k = 1, 2, ..., K$ is the switch index. The non-critical packets are scheduled to the lower priority queue $Q_k^L, k = 1, 2, ..., K$. In some special cases, some non-critical packets could suddenly become critical according to the current context. Recall, IoMT data that is collected from an IoMT application. Therefore, such packets will be scheduled in the high-priority queue with probability $1 - p_t$, where p_t is the probability of scheduling a sampling data packet to Q^L. In general, both critical (c) and non-critical (nc) packet arrivals are assumed to be poisson distributed, poisson traffic is used to obtain closed form results in M/G/1 queues, with rates λ_c and $\lambda_{nc} = \sum_{m=1}^{M} \gamma_m$, respectively.

Since there are two different types of traffic and consequently priority queues, the total rate of high-priority traffic λ_H and low-priority traffic λ_L is given by:

$$\lambda_H = \lambda_c + (1 - p_t) \cdot \lambda_{nc} \quad (1)$$

$$\lambda_L = p_t \cdot \lambda_{nc} \quad (2)$$

Network switches are assumed to operate following exponential distributed service time of mean $\overline{X}_i$ for node $i \in \{1, 2, \ldots, K\}$, and the two types of the IoMT traffic requires different service times. To maintain a guaranteed behavior for the real-time IoMT traffic, the expected delay experienced by each data packet since its transmission until it reaches the data sink must not exceed a predefined threshold T_{QoS}. Thus, if we let D be a random variable that stands for the end-to-end delay, then the probability of dropping a data packet at the application sever is given by:

$$P_{\text{blocking}} = \Pr\{D > T_{\text{QoS}}\} \quad (3)$$

In this case, our objective is to compute the maximum allowable hop-count K^* such that $\mathbb{E}[D] < T_{QoS}$, and estimate the packet dropping probability for a given number of hops in the network.

IV. Analysis

As mentioned earlier, our network consistent of K switches with two priority queues and, possibly, different service rate distributions. Initially, we are interested in obtaining the average waiting time $\overline{W_p}$ and second moment $\overline{W_p^2}$ for priority $p \in \{L, H\}$ that will be experienced by sampling data packets at each switch. Since the output of each queue is approximately poisson distributed process, we look into each switch in isolation and temporarily drop the switch index i. Thus, the waiting time of a high priority packet, denoted as W_H, is

$$W_H = \sum_{j=1}^{N_H} X_j + R, \quad (4)$$

where N_p Number of packets scheduled in the queue with priority p, $p \in \{L, H\}$, X_j is the service time of packet j and R is the residual time. From (4), using Little's formula, we obtain the average waiting time of the high priority queue $\overline{W}_H$ as,

$$\overline{W}_H = \frac{\overline{R}}{(1 - \rho_H)}, \quad (5)$$

where the first moment of this residual time is

$$\overline{R} = \frac{1}{2}\left((\rho_H + \rho_L) \cdot \frac{\overline{X^2}}{\overline{X}}\right) \quad (6)$$

To reiterate, $\rho_H = \lambda_H \cdot \bar{X}$ is the fraction of time a switch is serving high priority traffic. Thus, the second moment of waiting time for high-priority traffic:

$$\begin{aligned}\overline{W_H^2} &= \overline{N}_H \cdot \mathrm{Var(X)} + \mathrm{Var(R)} + \mathrm{Var(N_H)} \cdot \overline{\mathrm{X}}^2 \\ &\approx \overline{N}_H \cdot \mathrm{Var(X)} + \mathrm{Var(R)}, \end{aligned} \quad (7)$$

where

$$\begin{aligned}\overline{N}_H &= \lambda_H \, \overline{W}_H \\ \rho_H &= \lambda_H \cdot \overline{X} \\ \mathrm{Var}(R) &= \overline{R^2} - \overline{R}^2\end{aligned}$$

Using the law of total expectation, $\overline{R^2}$ is

$$\begin{aligned}\overline{R^2} &= (\rho_H + \rho_L) \cdot \overline{R^2} \\ &= \frac{1}{3}\left((\rho_H + \rho_L)\, \frac{\overline{X^3}}{\overline{X}}\right)\end{aligned} \quad (8)$$

From the above, we understand that if high priority traffic is limited, then high priority queues contain at most one packet almost all the time, and waiting time for high priority packets is chiefly due to residual time only.

For lower-priority traffic, the waiting time of a low priority packet can be expressed as

$$W_L = \sum_{j=1}^{N_H} X_j + \sum_{j=1}^{N_L} X_j + \sum_{j=1}^{W_L \cdot \lambda_H} X_j + R \quad (9)$$

Thus the average waiting time of the low priority queue denoted by $\overline{W}_L$ is,

$$\overline{W}_L = \frac{\overline{R}}{(1 - \rho_H)(1 - \rho_H - \rho_L)}, \quad (10)$$

and, the second moment is

$$\begin{aligned}\overline{W_L^2} &= \overline{R^2} + s \cdot \mathrm{Var}(X) \\ &+ s^2 \cdot \overline{X}^2 + 2\, s\, \overline{X} \cdot \overline{R} + q \cdot \overline{X}^2 \\ &\approx \overline{R^2} + s \cdot \mathrm{Var}(X) + s^2 \cdot \overline{X}^2 + 2\, s\, \overline{X} \cdot \overline{R}, \end{aligned} \quad (11)$$

where the coefficient s is defined as

$$s = \overline{N}_L + \overline{N}_H + \lambda_H \cdot \overline{W}_L$$

Here, q is variance of the number of packets in the three cases mentioned earlier and is assumed to be negligible in steady state, hence the last approximation follows.

A. Maximum Hop-count

The maximum allowable hop-count that respects the QoS delay constraint is given by

$$K^* = \arg\max_K \left\{ \sum_{i=1}^{K} \overline{T}_i \leq T_{QoS} \right\}, \tag{12}$$

where

$$\overline{T}_i = (1 - p_t) \cdot \overline{W}_{H,i} + p_t \cdot \overline{W}_{L,i} + \overline{\chi}_i \; + \; \tau_i, \tag{13}$$

where $\overline{W}_{H,i}$ and $\overline{W}_{L,i}$ are respectively given by (5) and (10) for each $i = 1, 2, \cdots K$, χ_i is the service time random variable at switch i, and τ_i is the propagation delay between switch $i-1$ and switch i.

In the special case where all switches are identical, we obtain the simpler expression:

$$K^* \approx \frac{T_{QoS}}{\overline{T}}, \tag{14}$$

with $\overline{T} = \overline{T}_i$, $i = 1, 2, \cdots K$. To draw qualitative insights, we note in the latter case that if traffic is limited, i.e. $\rho \approx 0$, then K^* is approximately given by the first order Taylor expansion:

$$K^* \approx \frac{T_{QoS}}{\overline{\chi} + \overline{\tau}} \; \cdot \; \left(1 - \rho \, \frac{\overline{R}}{\overline{\chi} + \overline{\tau}} \right) \tag{15}$$

Here, $\bar{\chi} + \bar{\tau}$ is the minimum average delay at a node regardless of traffic utilization. Hence, the impact of increasing traffic utilization in IoMT application is largely influenced by the ratio of residual service time $\overline{R} = \overline{X^2}/(2\overline{X})$ to such minimum average delay.

B. Blocking Probability

In order to obtain the average number of packets that exceed a predefined threshold T_{QoS}, we approximate the sum of all service times and waiting times experienced by a packet from source to sink by a normal distribution [6]. We also know the first and second moments of the random variables as derived earlier. We are interested to obtain the end-to-end delay $D = \sum_{i=1}^{K} W_i + \sum_{i=1}^{K} X_i$ for both high-priority traffic D_H and low-priority traffic D_L. Thus, we have

$$D_p \sim \mathcal{N}(\mu_p, \sigma_p), \tag{16}$$

where

$$\mu_p = \sum_{i=1}^{K} \overline{\chi}_i + \sum_{i=1}^{K} \overline{\omega}_{p,i}, \tag{17}$$

$$\sigma_p^2 = \sum_{i=1}^{K} \text{Var}(\chi_i) + \sum_{i=1}^{K} \text{Var}(\omega_{p,i}), \tag{18}$$

where χ_i denotes service time at node i and $\omega_{p,i}$ denotes waiting time at node i for traffic with priority $p \in \{L, H\}$.

Therefore, the final desired probability of arriving later than T_{QoS} is a weighted sum according to whether a data packet is scheduled in the high-priority queue or the low-priority queue:

$$\begin{aligned} \text{P}[\text{D} > T_{\text{QoS}}] \;\; &\approx \;\; \frac{1}{2}\,(1 - p_t) \left[1 - \text{erf}\left(\frac{1}{\sqrt{2}} \cdot \frac{T_{\text{QoS}} - \mu_{\text{H}}}{\sigma_{\text{H}}} \right) \right] \\ &+ \;\; \frac{1}{2}\, p_t \left[1 - \text{erf}\left(\frac{1}{\sqrt{2}} \cdot \frac{T_{\text{QoS}} - \mu_{\text{L}}}{\sigma_{\text{L}}} \right) \right] \end{aligned} \tag{19}$$

where $\text{erf}(\cdot)$ is the error function [7, Eq.(8.250.1)].

C. Service Placement

For the multimedia IoMT traffic, the objective QoS metrics are considered as an influencing factor for the multimedia QoE parameters. QoE can be defined as a function of the QoS provided by the network as follow:

$$QoE = f(QoS) \tag{20}$$

This can be done by considering quality models able to map objective measures (QoS) with the subjective measurements such as The Mean Opinion Score (MOS) [8]. Such subjective test results can be collected by the application operators and injected to the software-Defined-Networks (SDN) controllers that use both the QoS and QoE measures to set the polices in order to place the service near to the end users.

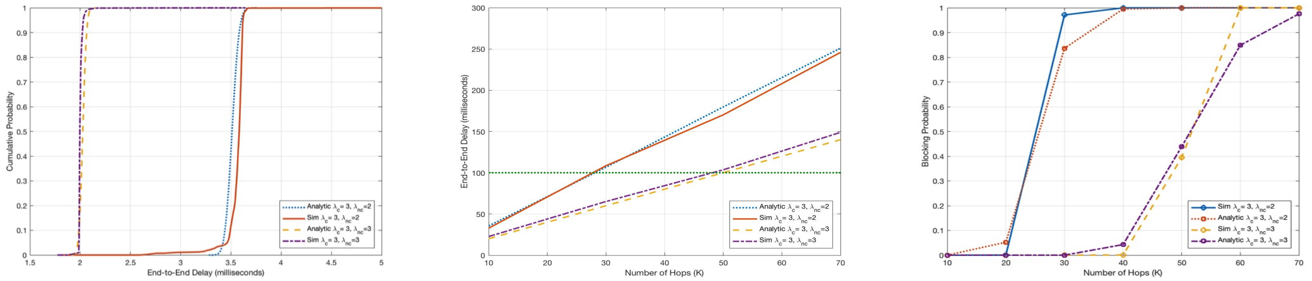
V. Performance Evaluation

In this section, the performance of the above framework is evaluated using MATLAB/Simulink. We will evaluate the QoS measurements and leave the subjective QoE tests for future works. For the analysis purpose of this work, we ignore the underlying communication technologies and use the abstract model.

Two types of IoMT applications are considered, critical and non-critical applications. For the real-time critical IoMT application, the end-to-end delay requirement of an Intelligent Transport System (ITS) is evaluated. Transportation Services such as road safety, intersection and traffic efficiency at both Urban or highways use smart cameras that distributed all over the targeted area to send warning traffic about collisions or dangerous situations. Therefore, the communication system has to operate with communication latency of less than 100 ms while ensuring high reliability [1]. By taking these requirements into consideration, we defined our end-to-end delay threshold T_{QoS} to be 100 milliseconds.

We simulated critical application that generates data at a poisson rate λ_c of one packet every 3 milliseconds. We also simulated non-critical application that produces poisson traffic with two different arrival rates λ_{nc} of one packet every 2 and 3 milliseconds. As described above, two measures played a major role in providing acceptable QoS requirements for critical applications, end-to-end delay, and maximum allowable network hop-count.

With regard to end-to-end delay, test results for one hop are shown in Figure 1a. The average end-to-end delay for the critical application is around 2 milliseconds when both



(a) End-to-End Delay for One Hop

(b) End-to-End Delay for Different Number of Hops

(c) Packet Dropping Probability

Fig. 1: End-to-End Delay and Packet Dropping Probability v.s Number of Hops

types of applications have an arrival rate of one packet every 3 milliseconds $\lambda_c = \lambda_{nc} = 3$. As the number of low priority packets increased, $\lambda_{nc} = 2$, the contention with the high priority data packets increased, this makes the average delay to be 3.5 milliseconds for the critical application.

In order to verify the effects of the number of hops on the end-to-end delay, a simple experiment is done while increasing the hop-count K from 10 to 70; Figure 1b demonstrates the average end-to-end delay for the critical application. Both analytic and simulation results are obtained. For the case of $\lambda_c = \lambda_{nc} = 3$, the average delay ranges between 20 and 60 milliseconds for 10 and 30 hops, respectively, that is still within the acceptable 100 milliseconds threshold T_{QoS}. The delay reaches the 100 milliseconds limit at the 50^{th} network hop. With non-critical packets arrival rate $\lambda_{nc} = 2$, the end-to-end delay easily hit the T_{QoS} at the 30^{th} hops with an average of 106 milliseconds. This gives us a good reason to define the limit on the number of hops a packet is allowed to traverse at the design stage of IoMT network infrastructure. Propagating such performance information toward the network edges can help to find the best route for the critical packets to the destination and meet the required end-to-end delay constraints.

Finally, the packet dropping probability as a function of K is plotted in Figure 1c. Both simulation and analytic results are displayed for the two different arrival rates of non-critical applications $\lambda_{nc} = 2$ and $\lambda_{nc} = 3$. When $\lambda_{nc} = 2$, the dropping probability is less than 1% for $K = 10$, 5% for $K = 20$ and escalated up to more than 80% for $K = 30$. Thus, 100% of the packets will have an end-to-end delay above the adequate limit of T_{QoS} when K is greater than 30 hops. Again with the case of less contention between critical and non-critical packets, $\lambda_{nc} = 3$, the packet dropping probability is below 1% for $K = 10$, $K = 20$, and $K = 30$. This probability rises up to 5% for $K = 40$, and then to 43% for $K = 50$ and continues to grow to more than 80% at 60^{th} and 70^{th} network hops. As shown from the Figures, our simulation closely matching the analytical results for all performance metrics.

Considering this results, it is critical for systems designers to analyze and quantify the inherent tradeoffs between QoS and the services placement. End-to-end delay and packet dropping are key metrics for the applications performance that need to meet strict threshold values. As a consequence, define the set of services within the application itself and deploy the required service dynamically toward the users at real time can be a potential candidate to fulfil the real-time critical requirements.

VI. Conclusions

End-to-end delay is an important parameter for IoMT real-time applications. The whole system will be broken down, and every incoming packet will be useless if the delay exceeds some predefined threshold. In this work, we present our analytical expressions for the maximum hop-count as well as the packet dropping probability for an IoMT application. We employed $M/G/1$ queues and used the end-to-end delay threshold for determining delayed packets. Our results can give an incite to network designers for determining the total number of hop-counts for their application. Using such information with the new technologies, such as software-Defined-Networks (SDN), can help to create and deploy distributed services near to the end users and therefore meet QoS and QoE requirements for different types of multimedia applications.

References

[1] P. Schulz, M. Matthe, H. Klessig, M. Simsek, G. Fettweis, J. Ansari, S. A. Ashraf, B. Almeroth, J. Voigt, I. Riedel *et al.*, “Latency critical iot applications in 5g: Perspective on the design of radio interface and network architecture,” *IEEE Communications Magazine*, vol. 55, no. 2, pp. 70–78, 2017.

[2] C. Long, Y. Cao, T. Jiang, and Q. Zhang, “Edge computing framework for cooperative video processing in multimedia iot systems,” *IEEE Transactions on Multimedia*, vol. 20, no. 5, pp. 1126–1139, 2018.

[3] M. Alaslani and B. Shihada, “Intelligent edge: An instantaneous detection of iot traffic load,” in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[4] W. Wang, Q. Wang, and K. Sohraby, “Multimedia sensing as a service (msaas): Exploring resource saving potentials of at cloud-edge iot and fogs.” *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 487–495, 2017.

[5] I. Adan and J. Resing, “Queueing theory. eindhoven university of technology, 180 pages, 2002,” 2010.

[6] S. Zahl, “Bounds for the central limit theorem error,” *SIAM Journal on Applied Mathematics*, vol. 14, no. 6, pp. 1225–1245, 1966.

[7] I. S. Gradshteyn and I. Ryzhik, *Table of Integrals, Series, and Products*, 7th ed. Amsterdam: Elsevier/Academic Press, 2007.

[8] D.-H. Shin, “Conceptualizing and measuring quality of experience of the internet of things: Exploring how quality is perceived by users,” *Information & Management*, vol. 54, no. 8, pp. 998–1011, 2017.