Lee, G

QoS Implementation with Triple-Metric Based Active Queue Management for Military Networks

Article

Article

QoS Implementation with Triple-Metric Based Active Queue Management for Military Networks

Gyudong Park [1], Byungchun Jeon [2] and Gyu Myoung Lee [3,*]

[1] Command and Control Systems PMO, Agency for Defense Development; iobject@add.re.kr
[2] Netvision Telecom Inc.; bcjeon@netvisiontel.com
[3] Faculty of Engineering and Technology, Liverpool John Moores University; g.m.lee@ljmu.ac.uk
* Correspondence: g.m.lee@ljmu.ac.uk; Tel.: +44 (0)151 231 2052

Abstract: For supporting Quality of Service (QoS) in a military network, applications of triple-metric priority of performance, importance, and urgency as well as autonomous and lightweight implementation are required. In the previous study, we analyzed a Korean military network's QoS implementation in the perspective of the triple-metric, and presented some improvements in the simplification of service classes of Differentiated Services (DiffServ). To extend the simplified DiffServ in the previous research, this paper proposes Active Queue Management (AQM) algorithms to process the traffic of each service class differently based on importance and urgency, and shows the feasibility through some experiments.

Keywords: QoS; DiffServ; Triple-Metric; AQM

1. Introduction

As military networks are operated in very harsh and dynamic environment, the risk of failures or malfunction is high, however rapid recovery and repair are limited [1]. Therefore, it is essential to apply Quality of Service (QoS) that differentiates according to priorities in a congestion situation in a military network. In addition, QoS implementation can be replaced with an optimization problem that selects or develops, as well as combines and tunes the appropriate priority criterion and processing methods.

In the QoS field for military networks, so called the triple-metric priority criterion of performance, importance and urgency is most widely cited and applied. Each priority is associated with a traffic type, a user or mission, and a timeliness. However, it is not easy to implement efficiently and effectively by applying these all three priorities.

In the previous study [2], we analyzed the QoS implementations of US military's DoDIN (*DoDIN: Department of Defense Information Networks*) [3][4] and Korean military's TICN (*TICN: Tactical Information Communication Networks*) [5] focusing on the triple-metric, and proposed several improvements for QoS of Korean tactical communication networks. We simplified the service class classification of Differentiated Services (DiffServ) [6] to four which is suitable for performance-based differentiated processing of traffic, suggested additional required criteria such as importance classification, and presented a prioritized traffic processing mechanism for each service class according to importance and urgency. In the performance analysis, we have demonstrated that the experimental results showed almost the same or better performance with no difference in performance under the same conditions. Furthermore, we also emphasized that autonomy and lightweight solutions are essential for the military network, especially for tactical services because its Disconnected, Intermittent, Limited (DIL) characteristic increases the risk of disconnection with remote services, and its Size, Weight and Power (SWaP) sensitivity limits the complicated and sophisticated processing.

This paper focuses on the extended capability of the simplified DiffServ using Active Queue Management (AQM) [7][8]. AQM enables to add special processing to each service class or queue of DiffServ. Therefore, in addition to performance-based differentiated traffic processing through DiffServ, it is necessary to implement importance-based and urgency-based differentiated traffic processing through AQM while meeting the demands for weight reduction and autonomy of the military network.

Considering the above requirements, this paper proposes the implementation of QoS for military networks based on the simplified DiffServ and the basic idea of AQM. In addition, this paper develops and presents differentiated traffic processing algorithms based on performance and urgency for each service class that can be implemented through the proposed AQM. It also shows their feasibility through some experiments.

The rest of this paper is organized as follows. In Section 2, we briefly explain related works. In Section 3, we explain the proposed AQM algorithms for the four service classes. Then, Section 4 presents experiments and their performance results. Finally, a conclusion is drawn in Section 5.

2. Related works

QoS implementation approaches can be divided into a flow-based approach and a class-based approach. And each is represented by Integrated Services (IntServ) [9] and Differentiated Services (DiffServ) [6]. Although IntServ and DiffServ are very old standards, they are still widely used due to their high degree of maturity. Most commercial equipment adopts DiffServ, which is more advantageous in terms of scalability, but the absolute guarantee of important flow-required performance of IntServ or flow-based QoS is an advantage that is difficult to give up. Therefore, a flow-based QoS approach is evolving into a standard named Time Sensitive Networking (TSN) [10][11][12][13] or Deterministic Networking (DetNet) [14][15][16]. In addition, there are studies that try to combine the two approaches [17].

It is a very popular research approach to further improve QoS by applying the latest trendy techniques to the good traditional techniques. Recently, big data or machine learning technology has been attempted in various fields. There are also efforts to support QoS more intelligently, especially in DiffServ [18].

The authors in [19] proposed a machine learning framework to dynamically further segment existing traffic classification mechanisms. Additionally, the authors in [20] proposed a machine learning-based service class classification for encrypted traffic.

The authors in [21] proposed an IOTA-based QoS guaranteed Flow system (IQF), a system that improves QoS by utilizing a blockchain or cryptocurrency technology. IOTA is a cryptocurrency designed specifically for Internet of Things (IoT).

The development of networks goes hand in hand with the development of applications. This is because an emerging network meets the needs of new applications and enables the emergence of new applications. Recently, with the advent of 5G, Software Defined Networking (SDN) is attracting attention, and SDN-based QoS control is one of the popular research topics [22].

Next-generation communication systems such as 5G must support new applications that require finer QoS. The authors in [23] proposed a flow-based QoS control and Context-oriented Transport (CoT) that understands application behavior and adapts to the dynamic state of the network. CoT is an end-to-end software solution that improves the underlying network capabilities.

A study to test and evaluate the performance of existing QoS algorithms for a specific network also has practical value. The authors in [24] tested three QoS algorithms for Multiprotocol Label Switching-Traffic Engineering (MPLS-TE) networks: Priority Queuing (PQ), First In First Out (FIFO), and Weighted Fair Queuing (WFQ). As another practical approach, there is also a study to apply QoS to an existing network that did not apply QoS. The authors in [25] proposed the QoS application extension for Information Centric Networking (ICN) [26].

Apart from QoS factors for DiffServ, additional factors like energy consumption and resource utilization are also important for improving overall network performance and supporting specific tactical applications with computing intensive features in military networks [27][28][29][30].

Most of the latest studies related to QoS are being carried out as improvements by applying the latest technology to the traditional QoS technology. This paper also pursues QoS improvement based on the proven existing technology, i.e., DiffServ. However, other studies assume a very stable commercial communication network and incorporate trendy technologies. In this regard, this paper aims to differentiate QoS with the triple-metric (i.e., performance, importance, and urgency) as an improvement in reliable and proven methods, focusing on lightweight and autonomous QoS technologies for military networks that can be very poor. The enhancement of the proposed solution needs to be continuously discussed in order to support additional requirements according to specific tactical applications in military networks.

3. Triple-metric based active queue management algorithms

Since the requirements for importance and urgency are different for each service class, the corresponding processing algorithm must also be implemented differently for each. In this regard, this paper develops all algorithms to operate simply when enqueuing and dequeuing considering performance, because processing in the queue is usually computationally expensive.

3.1. Service class classification

This paper cites and uses the service class classification result of the previous study [2]. In Table 1, the 'Control' service class illustrates the network control traffic, the 'Real-Time Multimedia' service class showcases voice or video traffic for real-time communication between users, the 'Low-Latency Data' service class includes data traffic for real-time information sharing or interaction between users and applications, and others are the 'Best Effort' service class.

Table 1. Service Classes.

Service Class	Examples	Note
Control	Network Control	-
Real-Time Multimedia	Telephony, Conferencing	Inelastic
Low-Latency Data	Chatting, Messenger, Web Application	Elastic
Best Effort	Others	Elastic

This service class classification integrated twelve service classes of the DiffServ standard into four. Based on the simplified service class classification, this paper proposes the highest QoS requirements of the integrated service classes, reflecting the QoS characteristics applicable to different types of traffic in each service class (see Table 2).

Table 2. Service class characteristics.

Service Class	Tolerance to		
	Loss	Latency	Jitter
Control	Low	Low	Very Low
Real-Time Multimedia	Very Low	Very Low	Very Low
Low-Latency Data	Low	Low	-
Best Effort	-	-	-

3.2. Control service class processing

All traffic of the control service class have high importance and urgency. Therefore, most DiffServ implementations allocate the control service class to Expedited Forwarding (EF) queue, and allocate sufficient bandwidth to the queue to prevent congestion. Therefore, this paper follows the same condition.

In case of network disconnection and degradation, it may not be possible to guarantee QoS even for the control service class. Therefore, resource waste can be reduced by applying AQM to drop the control service class that is not valid due to elapsed delivery time. However, this can be implemented simply, and the result can also be expected, we don't include it in the scope of this paper.

3.3. Real-time multimedia service class processing

In the critical and emergency situation, the real-time multimedia service class may increase rapidly for reporting, sharing, and command and control (C&C). In this case, additional priorities rather than performance are required for further differentiated processing. Since all real-time multimedia traffic have almost same urgency, importance based differentiated processing is appropriate. For this, this paper allocates the real-time multimedia service class to Assured Forwarding 4 (AF4) queue, and applies the following AQM algorithm (Algorithm 1).

Algorithm 1: ***Real-time multimedia service class processing***

```
line becomes idle
if queue is not empty then
  dequeue;
  if length of queue <= min_threshold then
    send;
  else if importance of the packet == highest then
    send;
  else if importance of the packet >
        importance of last packet in queue then
    send;
  else
    drop;

new packet is arrived
if queue is empty && line is idle then
  send;
else if queue is full then
  drop;
else if length of queue <= max_threshold then
  enqueue;
else if importance of the packet == highest ||
       importance of the packet >
       importance of last packet in queue then
  enqueue;
else
  drop;
```

The Algorithm 1 aims to support importance by mission priority while differentiating the performance by service class. The packet importance used in this algorithm follows the importance classification criterion of the previous study as shown in Table 3. This classification is applicable to all types of traffic such as voice, video, and data.

Table 3. Importance Level.

Level	Telephony/Conferencing	Data
FO (Flash Override)	Commander	Commander/Emergency
F (Flash)	Survival related	Operation Supporting
I (Immediate)	Security related	Mission Supporting
R (Routine)	Official	Administrative

This algorithm uses two different queue length thresholds for dequeuing and enqueuing. The threshold (i.e., min_threshold) used in the dequeue process is set as follows. This value can be a criterion for determining the congestion of real-time multimedia traffic.

$$\text{min\_threshold} = TS \times BR\_AF4 \times TD\_RT \,/\, N_{hops} \quad (1)$$

where *TS* is the transmission speed, *BR_AF4* is the bandwidth allocation ratio for AF4 queue in a switch node, *TD_RT* is the tolerable delay of real-time multimedia, and N_{hops} is the number of hops in the route. *TS* depends on the size of the packet, how long the packet is. Apart from relatively static information under the given network condition, the tolerable delay (*TD_RT*) is considered to reflect dynamic traffic characteristics (e.g., jitter, etc.) of the real-time multimedia service class.

If queue length is shorter than min_threshold, all packets in the queue can be delivered within tolerable delay, so there is no need to apply special processing. However, when the queue length becomes longer than min_threshold, this algorithm performs importance based differentiated traffic processing. In this case, this algorithm sends the highest importance packet and more important packets than the previous one, and drops others. In this way, this algorithm tries to guarantee the QoS of more packets of higher importance and considers relative importance between packets. And the other threshold used in dequeuing process, max_threshold, is set as follows.

$$\text{max\_threshold} = TS \times BR\_AF4 \times TD\_RT \,/\, HIPR \,/\, N_{hops} \quad (2)$$

where *HIPR* is the highest importance packet ratio in the queue.

If the queue length becomes longer than max_threshold, even the packet of highest importance cannot meet its QoS requirements. In this case, this algorithm drops all packets except most and more important packets than previous packets in the queue to strongly suppress the queue length increase.

If many packets of a real-time voice or video traffic are dropped, the voice call or video conference will be disconnected naturally. And this can have a positive effect to decrease the congestion of the real-time multimedia service class.

3.4. Low-latency data serivce class processing

The low-latency data service class includes traffic for fast and accurate information sharing and real-time interaction between users and applications. In case of critical situation low-latency data traffic can also be overwhelmed. In addition, the low-latency data service class also requires quite high level of performance. Therefore, differentiated processing based on urgency is not appropriate for the low-latency data service class, and importance-based differentiating is required.

The many delays of low-latency are caused by humans. Therefore, most users are quite generous to occasional short delays by communication networks. However, the failure of critical information delivery can be a significant problem. Therefore, this paper allocates the low-latency data service class to Assured Forwarding 2 (AF2) queue, and applies the following AQM algorithm (Algorithm 2) considering the reliability of the important traffic further.

Algorithm 2: ***Low-latency data service class processing***

```
new packet is arrived
if queue is empty && line is idle then
  send;
else if queue is full then
  drop;
else if length of queue < t1
  enqueue;
else if length of queue >= tn &&
      importance of the packet >= n then
  enqueue;
else
  drop;

line becomes idle
if queue is not empty then
  dequeue; send;
```

Instead of two different queue length thresholds (i.e., max_threshold and min_threshold) in the Algorithm 1, the Algorithm 2 uses different thresholds according to the importance of the packets. Therefore, the first threshold is set as follows.

$$1_{st}\_threshold = TS \times BR\_AF2 \times TD\_lld \, / \, N_{hops} \quad (3)$$

where BR_AF2 is bandwidth allocation ratio for AF2 queue, and TD_lld is tolerable delay of low-latency data.

If the queue length is shorter than 1_{st}_threshold, all packets in the queue can be delivered within tolerable delay. Therefore, there is no need to apply special processing. However, when the queue length becomes longer than 1_{st}_threshold, this algorithm drops the least important packets without enqueuing them. This paper sets subsequent i_{th}_threshold as follows. Also, the i_{th}_threshold should always be greater than the $(i\text{-}1)_{th}$ threshold.

$$I_{th}\_threshold = TS \times BR\_AF2 \times C \times I \times TD\_lld \, / \, N_{hops} \quad (4)$$

where C is the correction factor which is assigned to the queue. The correction factor can be adjusted by the traffic ratio of the queue.

If the queue length is longer than the i_{th}_threshold, this algorithm drops all packets below the i_{th} importance without enqueuing them. Additionally, this paper does not apply a special algorithm for dequeuing. This algorithm tolerates some delays to ensure the reliability of important traffic.

3.5. Best effort service class processing

Some traffic have delivery deadline flags and it becomes worthless when it exceeds the deadline. Therefore, a packet with short remaining time to the delivery deadline has high urgency. In addition, forwarding packets that have already exceeded or will exceed the delivery deadline waste network resources.

As all control, real-time multimedia, and low-latency data service classes have a very short delivery deadline, differentiated processing based on urgency may be impossible and meaningless. However, the best effort service class requires relatively low performance and can have a different delivery deadline depending on the application. Therefore, it is possible to apply urgency-based differentiated processing to the best effort service class. Accordingly, this paper allocates the best effort service class to Default Forwarding (DF) queue, and applies the following AQM algorithm (Algorithm 3).

Algorithm 3: ***Best effort service class processing***

```
new packet arrived
if new packet with delivery_deadline flag &&
  remain time of new packet
  < expected latency then
  drop;
else if queue is empty && line is idle then
  send;
else if queue is full then
  drop;
else
  enqueue;

line becomes idle
if queue is not empty then
  dequeue;
  if remain time of the packet <
      expected latency then
    drop;
  else
    send;
```

Apart from the differentiated performance by service classes and the importance by mission priority, the Algorithm 3 supports the urgency by delivery time, especially any urgency which is associated to time constraints. This algorithm avoids wasting network resources by dropping packets which exceeded or will exceed the delivery deadline.

On the other hand, not all traffic have a delivery deadline. Therefore, we need a way to tell whether a packet has a delivery deadline or not. For this, this paper uses 2 unused bits of Differentiated Services Code Point (DSCP) [31] in the packet header (see Table 4).

Table 4. Time limit codes.

Delivery deadline	Code
Yes	00
No	01

Finally, this paper proposes a method of giving remaining time of delivery in seconds by utilizing Time to Live (TTL) field of the packet header. Currently, the TTL field is used to prevent packets falling into an infinite loop by setting the maximum number of hops. However, as the name "Time To Live" suggests, its original use was to impose a remaining time of delivery by Nagle in 1987 [32]. This paper proposes to utilize the TTL field for both purposes.

4. Experiments

4.1. Simulation environment

Figure 1 shows switch node configuration of a military network for setting up the simulation environment of our entire study. This paper uses a part of that environment. We implemented our proposed simulation model (see Figure 2 for details) in each switch node of the military network supporting each military device with the traffic generator.

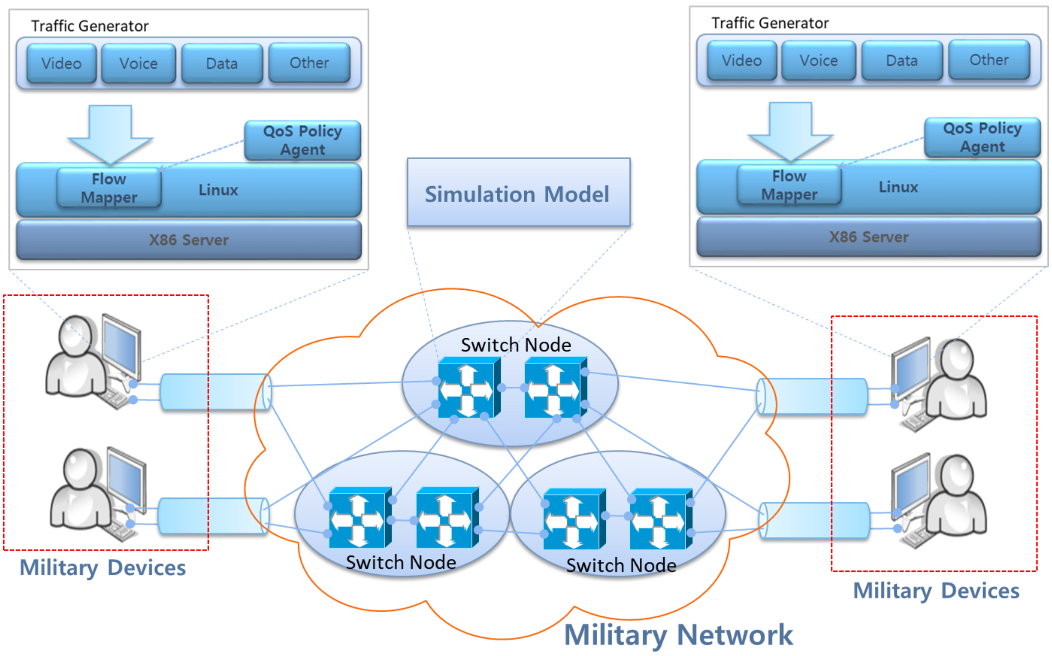


Figure 1. Switch node configuration of a military network.

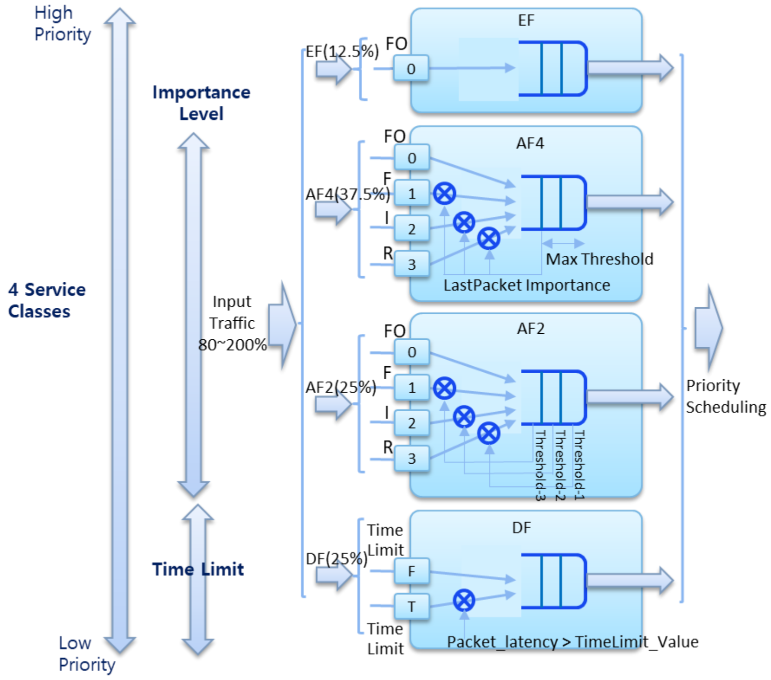


Figure 2. Configuration of the simulation model.

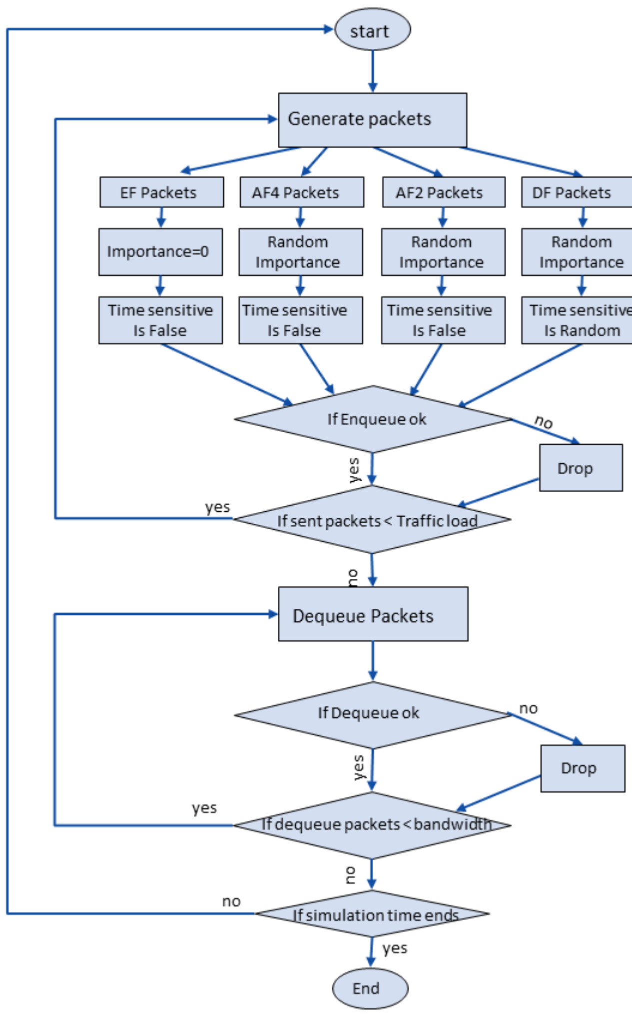


Figure 3. QoS implementation with the triple-metric based active queue management.

For configuration of the simulation model as shown in Figure 2, this paper considered the following conditions.

- Bandwidth: 2048Kbps, 256byte packet;
- Service Class Traffic Combination: EFRate 12%, AF4Rate 38%, AF2Rate 25%, DFRate 25%;
- Queue Size: EFQueueSize 1000, AF4QueueSize 1000, AF2QueueSize 1000, DFQueueSize 1000;
- Threshold: 1_{st}_threshold 0.25, 2_{nd}_threshold 0.51 and 3_{rd}_threshold 0.76 of Queue size.

To implement the proposed triple-metric based AQM, packets with four different service classes are simultaneously generated with different ratios. They have different priorities, importance levels and time-limit flags. As shown in Figure 3, the importance of all EF traffic is 0 (high priority). The importance is randomly allocated between 0 and 3 to the AF4 and AF2 traffic. In case of the DF traffic, time sensitive information is randomly allocated. Then, each switch node processes individual packets according to the proposed algorithms.

In this experiment, importance is applied to the EF, AF4, and AF2 traffic, requiring real-time or near real-time delivery, and a delivery deadline is applied to the DF traffic, requiring efficient delivery. The followings show the details of importance and delivery deadline assignment policy.

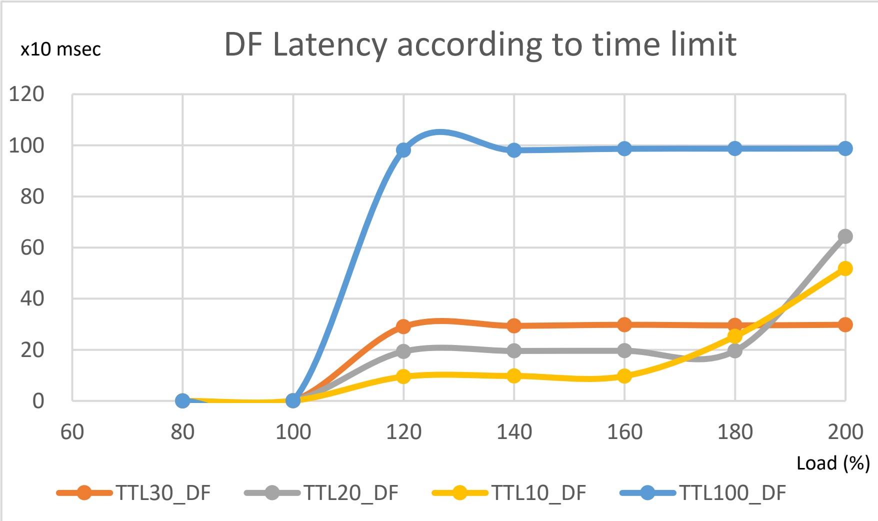
- The highest importance (FO) is assigned to the EF traffic;
- Four importance values (FO, F, I, R) are randomly assigned to the AF4 and AF2 traffic;
- Delivery time limit flags (true, false) are randomly assigned to the DF traffic. A false delivery time limit flag is fixedly assigned to the EF, AF4, and AF2 traffic.

For the experimental analysis of the proposed triple-metric based AQM algorithms to support performance by service class, importance by mission priority, and urgency by delivery time simultaneously, we aim to focus on performance of the AF4, AF2 and DF traffic according to importance and urgency while guaranteeing high performance of the EF traffic. In this regard, we applied importance based differentiated processing (Algorithm 1 in section 3.3) for the AF4 traffic and different loss thresholds considering the reliability (Algorithm 2 in section 3.4) for the AF2 traffic while satisfying the delay characteristics. In addition, we applied urgency-based differentiated processing (Algorithm 3 in section 3.5) for the DF traffic. The aim of the experimental analysis is to demonstrate the outstanding characteristics of the optimized algorithms (Algorithms 1, 2 and 3) to the simplified DiffServ.

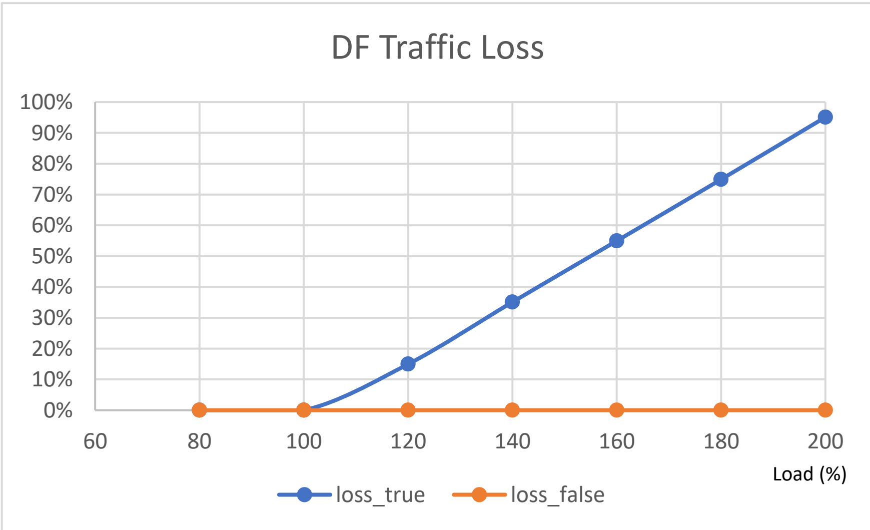
4.2. Performance of non real-time traffic (DF)

By allocating importance and delivery deadlines to the DSCP, traffic with an imminent delivery deadline can be delivered firstly. In the case of traffic within the delivery time limit of 1 (true), the traffic is dropped when the delay exceeds the specified time.

Figure 4 (a) shows the latency characteristics according to the delivery time limit value by changing the DF traffic load to 80-200% with only the DF traffic as input. As the input load increases, so does the traffic propagation delay. Traffic for which the delivery time limit flag is true is dropped when the delay exceeds the specified time. Therefore, the delay of the transmitted traffic without being dropped can be maintained within the specified time.



(a)



(b)

Figure 4. Performance results of the DF traffic: (a) Latency performance of the DF traffic; (b) Loss performance of the DF traffic.

Figure 4 (b) shows the difference between when the delivery time limit flag is set (1, the traffic that has passed the specified time is dropped) and there is the false delivery time limit flag (0). When the input load exceeds 100%, it is dropped by the amount of excess traffic. As shown in Figure 4 (b), traffic within the delivery time limit flag of 1 is dropped. The proposed method shows that the maximum propagation delay can be limited within a certain value even for traffic of the best effort service class.

Through this experiment, it was confirmed that the suggested algorithm works well. By dropping invalid packets which have passed the delivery deadline, we were able to avoid wasting resources and deliver more valid packets.

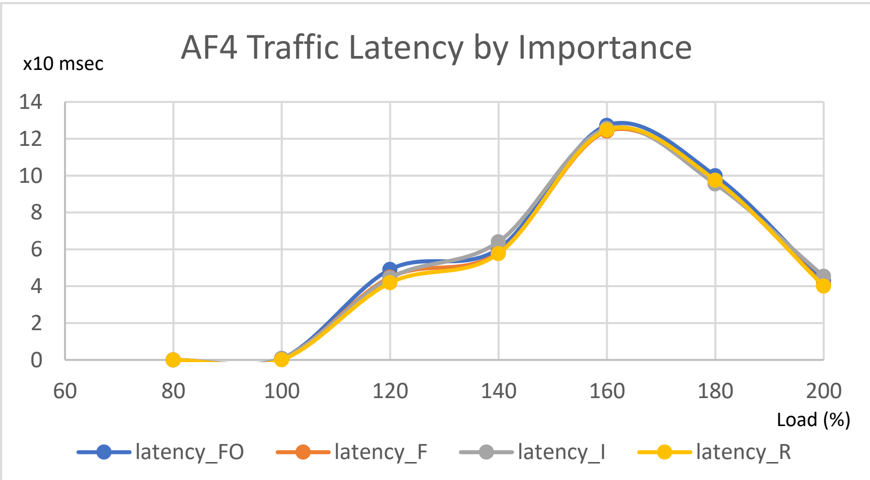
4.3. Performance of real-time traffic (AF4)

For the F4 traffic, the processing method is determined according to the queue length and importance. Figure 5 (a) confirms the result while changing the load to 80-200% with only the AF4 traffic as input, and randomly assigns four levels of importance (FO, F, I, R). In this result, when the load exceeds 100%, transmission delay occurs, however, when the queue length exceeds the threshold, a drop is made so that the delay does not increase excessively. A typical maximum tolerable latency for multimedia traffic is around 150 ms [33], and it can be seen that all traffic latencies in the experiment are less than 150 ms.

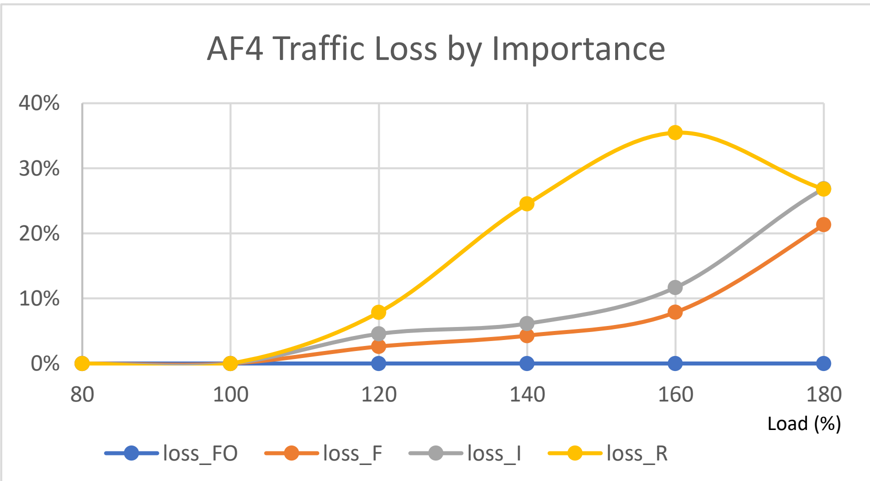
Figure 5 (b) shows the AF4 traffic loss characteristics with four levels of importance randomly assigned. Since the FO traffic with the highest importance does not drop regardless of the queue length, the loss value is maintained at 0, and it can be seen that the loss rate of the F, I, R traffic having the remaining three importance varies according to the loss threshold.

As the load exceeds 100%, R traffic-oriented loss occurs, and the F, I traffic loss rapidly increases around 160%.

From this result, in the case of the AF4 traffic, delay differentiation by importance hardly appears and affects loss differentiation. Therefore, it can be seen that it is effective for the bounded latency of traffic of the real-time multimedia service class and loss-free delivery of high-importance traffic in case of overload by considering both the queue length threshold and the importance within the same service class.



(a)



(b)

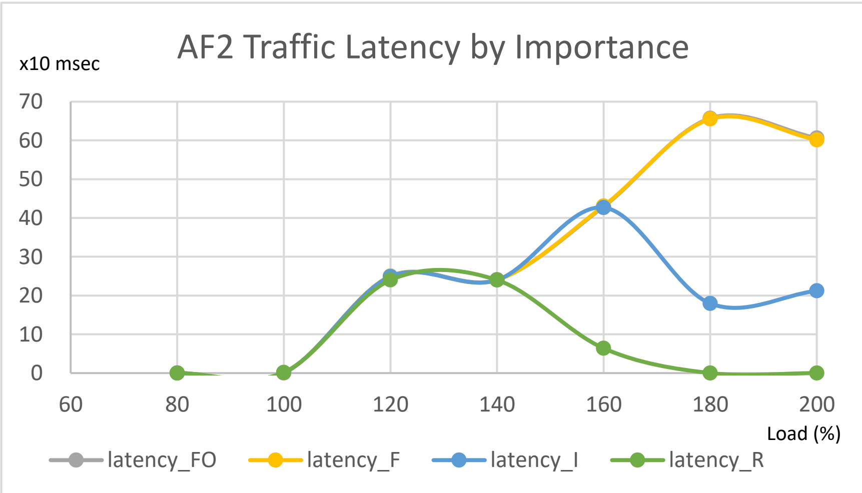
Figure 5. Performance results by importance of the AF4 traffic: (**a**) Latency performance by importance of the AF4 traffic; (**b**) Loss performance by importance of the AF4 traffic.

4.4. *Performance of low-latency traffic (AF2)*

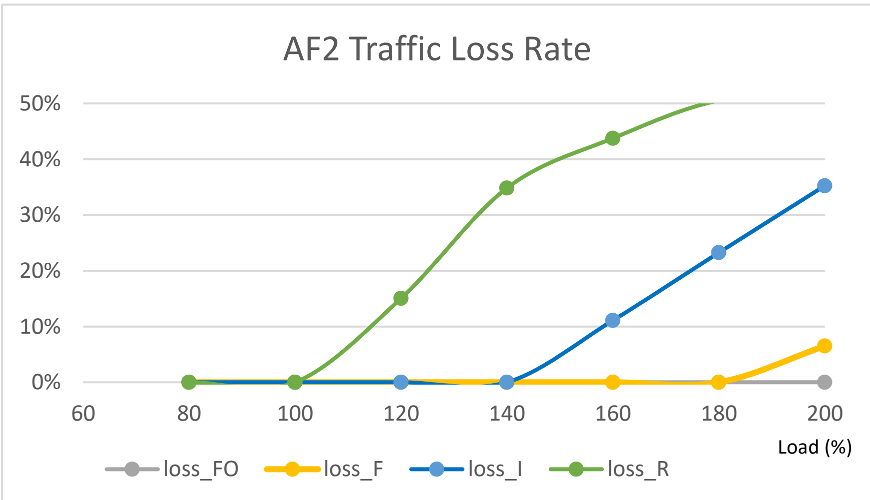
Figure 6 (a) confirms the result while changing the load to 80-200% with only the AF2 traffic as input, and randomly assigns four levels of importance (FO, F, I, R). When the load exceeds 100%, the delay increases rapidly. Even with the different loss rates for each importance, the delay is up to 140%. At input load above 133%, the traffic with R importance is mostly dropped, so latency appears to be reduced. The traffic with I importance is also lowering the average delay due to excessive loss at 180% or higher.

The AF2 traffic randomly assigns four importance, so the following results are expected:

- 0~100%: there should be no loss of traffic;
- 100~133.3%: only part of the R traffic is lost;
- 133~200%: 100% of the R traffic is lost and some of the I traffic is lost;
- 200% or more: some of the F traffic is lost.



(a)



(b)

Figure 6. Performance results by importance of the AF2 traffic: (a) Latency performance by importance level of the AF2 traffic; (b) Loss performance by importance level of the AF2 traffic.

Consistent with the above expectation, from Figure 6 (b), it can be seen that as the load increases, low-importance traffic loses firstly, and the F traffic starts to lose at 1800%

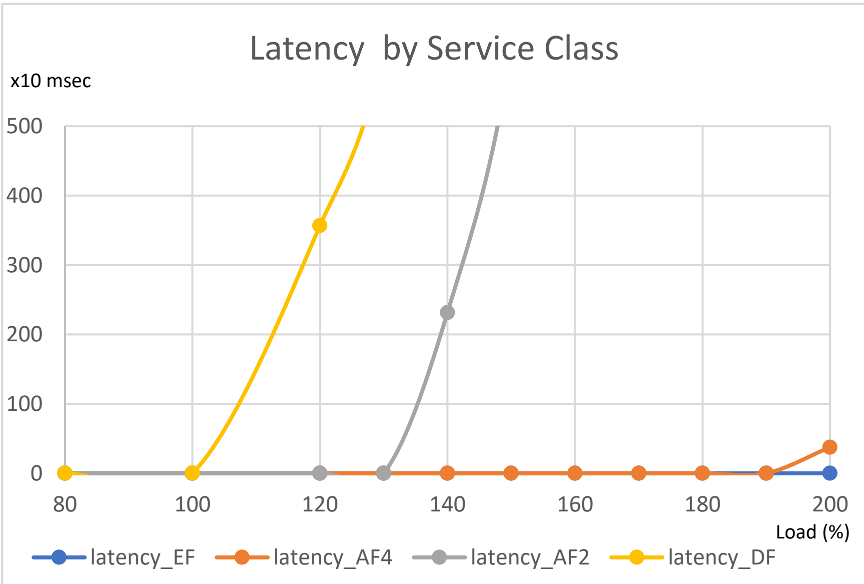
load. It's earlier than our expectation. The cause is that there are packets with various importance in the queue already, and our algorithm gives more priorities to the packets in the queue than the new arriving packets.

Therefore, for the AF2 traffic, it affects delay and loss differentiation by importance. It has a clear effect on loss-free delivery of high-priority traffic when overloaded within the same service class.

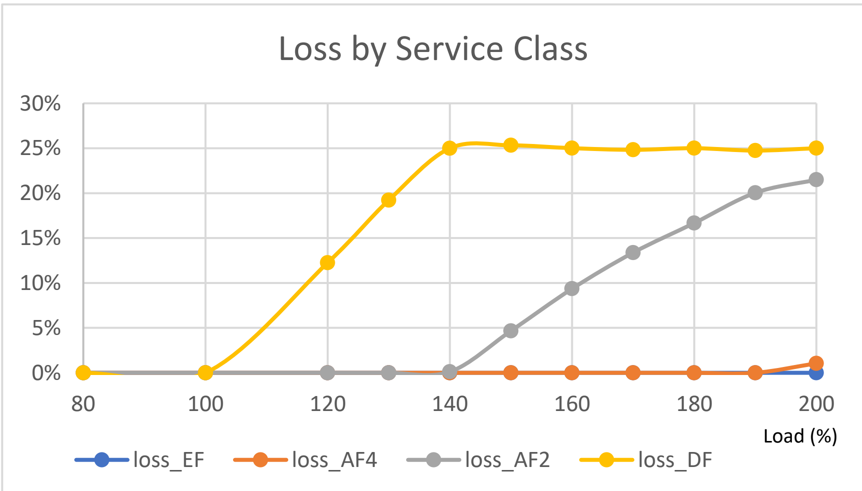
4.5. Performance of latency and loss by service class

In previous sections 4.2, 4.3 and 4.4, we have demonstrated performance behaviors of individual service classes, i.e., the DF, AF4 and AF2 traffic respectively. This section shows the performance analysis results for delay and loss for each service class when four classes of traffic are mixed as we have explained with the Figures in section 4.1.

Figure 7 (a) shows that when the load exceeds 100% in traffic environments with the four service classes, the delay increases sequentially from the low priority traffic. For the DF and AF2 traffic, the delay increases when the input load exceeds 100% and 180%, respectively, and the highest priority EF and AF4 traffic maintains low delay up to 200% load.



(a)



(b)

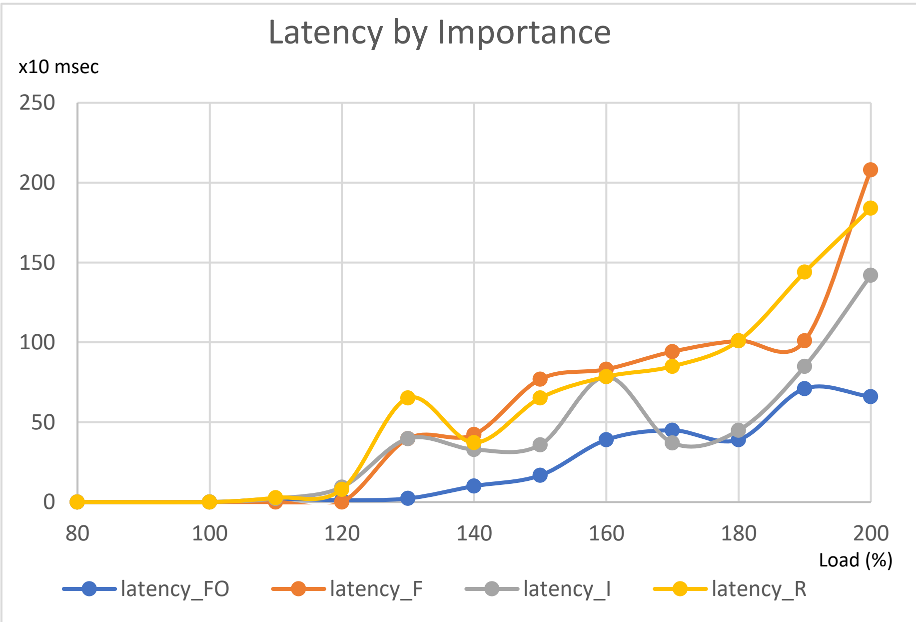
Figure 7. Performance results by service class for the EF, AF4, AF2 and DF traffic: (**a**) Latency performance by service class; (**b**) Loss performance by service class.

As mentioned earlier in the simulation environment, the following behavior is expected because the ratio of input traffic is EF (12%), AF4 (38%), AF2 (25%), and DF (25%).

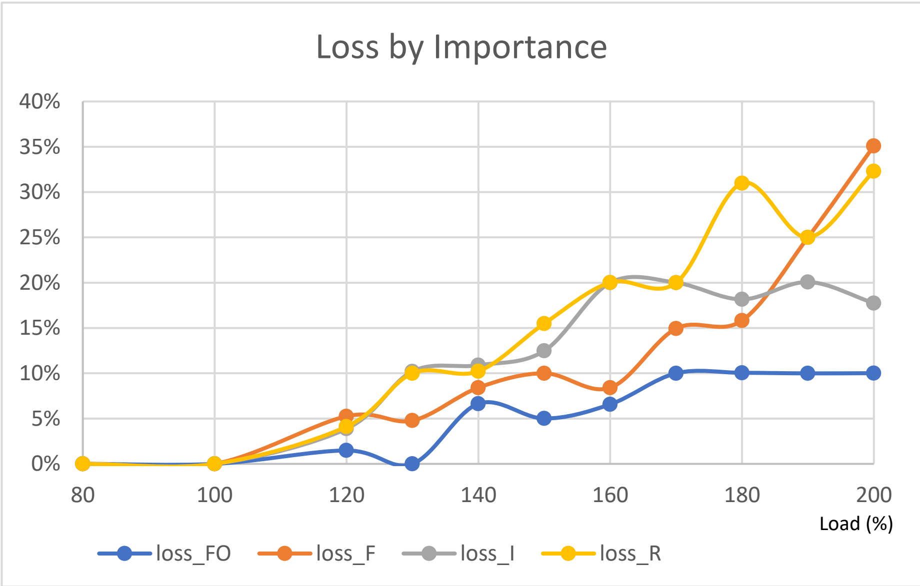
- Loss of the DF traffic occurs when input exceeds 100%;
- Loss of the AF2 traffic occurs when input exceeds 133.3%;
- Loss of the AF4 traffic occurs when input exceeds 200%.

Figure 7 (b) shows that, consistent with the above prediction, the loss of the DF traffic increases when the load exceeds 100%, and the loss of the AF2 traffic occurs when the load exceeds 140%. When the total load reaches 200%, the sum of the EF and AF4 traffic, which are the highest priorities, becomes 100%, so it is the point at which loss begins (some loss occurs).

Next, we look at the delay and loss characteristics according to the importance of each service class. In Figure 8 (a), the delay increases as the traffic load increases, but the delay of the FO traffic with high importance increases slowly.



(a)



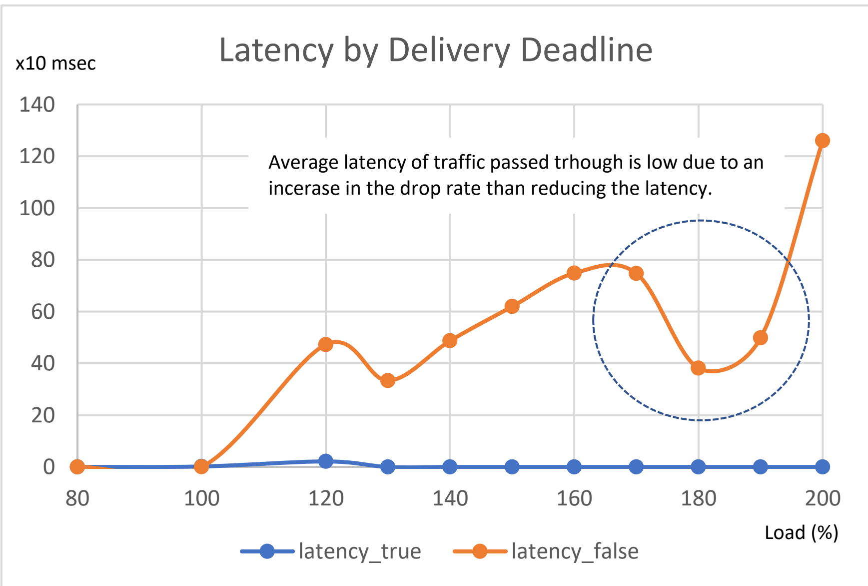
(b)

Figure 8. Performance results by importance for the EF, AF4, AF2 and DF traffic: (**a**) Latency performance by importance level; (**b**) Loss performance by importance level.

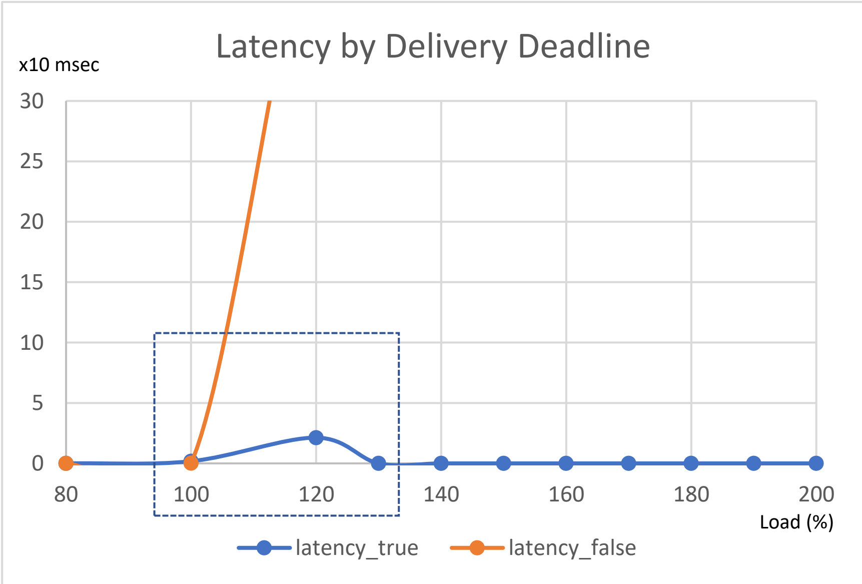
Figure 8 (b) shows that when the load exceeds 100%, the loss of traffic of all importance occurs. Also, the higher the importance, the lower the traffic drop rate.

Finally, the results of performance analysis on traffic differentiation according to the delivery time are reviewed.

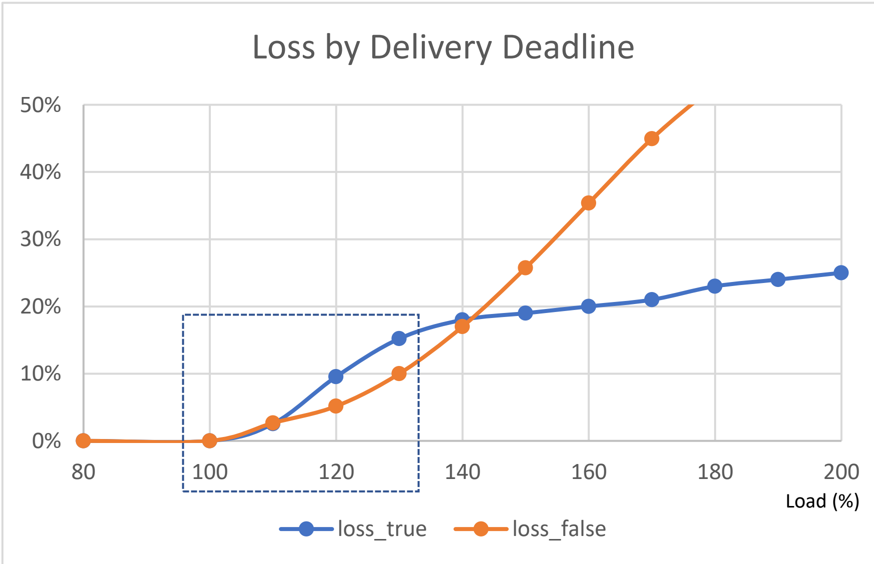
At 100% load, the DF traffic starts to lose, and at 133% load, 100% of the DF traffic will be dropped. Therefore, the meaningful period in Figure 9 (a) is the 100~133% load as highlighted with the rectangular box in Figure 9 (b). Since the delivery time limit is 0 (false) for all EF, AF4, and AF2 traffic and 50% of the DF traffic, a delay at a load of 133% or more represents the average of traffic delays of the four service classes. Traffic within the delivery time limit of 1 (true) is 50% of the DF traffic, and the delay is maintained within a certain level in the meaningful range of 100 to 133%, and it seems to converge to 0 because the amount of delivery is too small in the period where the loss is excessive.



(a)



(b)



(c)

Figure 9. Performance results of delivery deadline: (a) Latency performance by delivery time limit flags; (b) the meaningful period of (a); and (c) Loss performance by delivery time limit.

That is, since only half of the DF traffic has the delivery time of 1 (true), it is responsible for maintaining the delay within a certain range only in the 100-133% load period, and the delay appears as "0" at a load of 133% or more because all traffic have been dropped.

As mentioned earlier, the traffic within the delivery time limit of 0 (false) is the traffic that includes all EF, AF4, and AF2 traffic and 50% of the DF traffic (occupies 75% of the total traffic), and only the DF traffic in the 100-133% load range. If a drop occurs and the delivery delay exceeds the time limit, traffic within the delivery time limit of 1 (true) is dropped first. At a load of 133% or more, the AF2 traffic starts to drop, so the drop of traffic within the delivery time limit of 0 (false) gradually increases. Traffic within the delivery time limit of 1 (true) occupies 12.5% of the total traffic, so if the total load is 200%, a 25% drop occurs.

The last result shown in Figure 9 (c) is a case where the traffic with the four service classes are mixed, and differentiation according to time limit appears in the 100-133% load period. It can be seen that the drop according to the delivery time limit is controlled in the DF traffic, which is the lowest priority, so that it is effective in delivering the traffic within a time limit even when overloaded.

5. Conclusions

The simplified DiffServ and triple-metric based AQM-based QoS implementation proposed in this paper improves the delivery characteristics required in the military network by the importance and delivery time limit flags.

Among the four service classes, four levels of importance were assigned to the AF4 and AF2 service classes. In case of overload, by assigning different loss thresholds for each importance, high-importance traffic could be delivered first. It was verified that it is possible to guarantee the delivery of the most important traffic without degrading the delay characteristics.

For the DF traffic, a method of dropping traffic exceeding the delivery time limit was applied in case of overload. It was verified that the waste of network resources can be reduced and as a result, the delay of transmitted traffic can be maintained within a certain value.

This can be applied as a method to ensure the delivery of important traffic even when it is overloaded in the military network. Apart from the triple-metric, we believe that other factors such as energy consumption and resource utilization are also important. For future research, we will further focus on the enhancement of the proposed solution considering other QoS factors for some specific tactical applications with different requirements while improving overall network performance.

Author Contributions: Conceptualization, G.Park.; methodology, G.Park. and B.Jeon.; validation, B.Jeon. and G.M.Lee.; writing—original draft preparation, G.Park.; writing—review and editing, G.Park., B.Jeon. and G.M.Lee. All authors have read and agreed to the published version of the manuscript.

Acknowledgments: This work was supported by the Agency For Defense Development Grant funded by the Korean Government (UG200084ED).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. K. Vesa and M. Syrjänen, "Modelling Utility of Networked Services in Military Environments," *2019 International Conference on Military Communications and Information Systems (ICMCIS)*, **2019**, pp. 1-6.
2. Gyudong Park, Hocheol Jeon, Gyu Myoung Lee, Byungchun Jeon. (**2022**). A Study on Implementation and Improvement of Triple-Metric Based QoS for Military Networks. *The Journal of Korean Institute of Communications and Information Sciences*, 47(7), 1025-1035.
3. CJCSI 6211.02D, Defense Information Systems Network (DISN) Responsibilities, JCS J-6, Jan. **2015**.
4. DoD Architecture Framework 1.5, DoD, Apr. **2007**.
5. E. Choi, B. Lim, J. Kim, Y. Kim, H. Choi, B. Kim, and S. Nam, "A study on measures to ensure the information exchange between weapon systems on the All-IP tactical network," Ajou Univ., **2014**.
6. S. Blake, et. al., An Architecture for Differentiated Services, IETF RFC 2475, December **1998**.
7. R. Adams, "Active Queue Management: A Survey," in *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1425-1476, Third Quarter **2013**.
8. S. Jung, J. Kim and J. -H. Kim, "Intelligent Active Queue Management for Stabilized QoS Guarantees in 5G Mobile Networks," in *IEEE Systems Journal*, vol. 15, no. 3, pp. 4293-4302, Sept. **2021**.
9. R. Braden, et. al., Integrated Services in the Internet Architecture: an Overview, IETF RFC 1633, June **1994**.

10. J. Joung and J. Kwon, "Zero Jitter for Deterministic Networks Without Time-Synchronization," in *IEEE Access*, vol. 9, pp. 49398-49414, **2021**.
11. Z. Satka, M. Ashjaei, H. Fotouhi, M. Daneshtalab, M. Sjödin and S. Mubeen, "QoS-MAN: A Novel QoS Mapping Algorithm for TSN-5G Flows," *2022 IEEE 28th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*, **2022**, pp. 220-227.
12. J. Xin, S. Xu, H. Zhang and S. Xiong, "A review of Time Sensitive Communication Technology," *2022 International Conference* on Computer Engineering and Artificial Intelligence (ICCEAI), **2022**, pp. 844-848.
13. L. Thomas, A. Mifdaoui and J. -Y. L. Boudec, "Worst-Case Delay Bounds in Time-Sensitive Networks With Packet Replication and Elimination," in *IEEE/ACM Transactions on Networking*, **2022**.
14. H. Yu, T. Taleb and J. Zhang, "Deterministic Latency/Jitter-Aware Service Function Chaining Over Beyond 5G Edge Fabric," in *IEEE Transactions on Network and Service Management*, vol. 19, no. 3, pp. 2148-2162, Sept. **2022**.
15. J. Prados-Garzon, T. Taleb and M. Bagaa, "Optimization of Flow Allocation in Asynchronous Deterministic 5G Transport Networks by Leveraging Data Analytics," in *IEEE Transactions on Mobile Computing*, **2021**.
16. A. Nasrallah et al., "Ultra-Low Latency (ULL) Networks: The IEEE TSN and IETF DetNet Standards and Related 5G ULL Research," in *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 88-145, First quarter **2019**.
17. J. Liu, "Design and Implementation of Vo IPQoS Model Combining IntServ and DiffServ Based on Network Processor IXP2400," 2021 7th Annual International Conference on Network and Information Systems for Computers (ICNISC), **2021**, pp. 60-64.
18. D. Aureli, A. Cianfrani, A. Diamanti, J. M. Sanchez Vilchez and S. Secci, "Going Beyond DiffServ in IP Traffic Classification," *NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium*, **2020**, pp. 1-6.
19. Davide Aureli, Antonio Cianfrani, Marco Listanti, Marco Polverini, Stefano Secci, Augmenting DiffServ operations with dynamically learned classes of services, Computer Networks, Volume 202, **2022**, 108624, ISSN 1389-1286.
20. Huang, Y.-F.; Lin, C.-B.; Chung, C.-M.; Chen, C.-M. Research on QoS Classification of Network Encrypted Traffic Behavior Based on Machine Learning. Electronics **2021**, 10, 1376.
21. H. Masaki, K. Nguyen and H. Sekiya, "A QoS-guaranteed System with Software Defined Networking and Micropayment," 2021 26th IEEE Asia-Pacific Conference on Communications (APCC), **2021**, pp. 94-98.
22. J. C. da Cruz de Lima and E. B. Rodrigues, "Quality of Service Control in Software-Defined 5G Networks," 2021 IEEE International Conference on Automation/XXIV Congress of the Chilean Association of Automatic Control (ICA-ACCA), **2021**, pp. 1-6.
23. J. M. Ppallan, K. Arunachalam, S. S. Gantha, S. Jaiswal, S. Song and A. Nigam, "A Method for Enabling Context-Awareness at Transport Layer for Improved Quality-of-Service Control," in IEEE Access, vol. 9, pp. 123987-123998, **2021**.
24. K. A. Bozed, A. M. Elbeskri, J. R. A. Zreg and A. R. Zerek, "Investigation the Performance Effect of QOS in MPLS-TE Network," 2021 IEEE 2nd International Conference on Signal, Control and Communication (SCC), **2021**, pp. 361-367.
25. Jessica McCarthy, S.R. Chaudhry, Perumal Kuppuudaiyar, Radhika Loomba, Siobhan Clarke, QoSA-ICN: An information-centric approach to QoS in vehicular environments, Vehicular Communications, Volume 30, **2021**, 100351, ISSN 2214-2096.
26. M. Rayani, A. Ebrahimzadeh, R. H. Glitho and H. Elbiaze, "Ensuring Profit and QoS When Dynamically Embedding Delay-Constrained ICN and IP Slices for Content Delivery," in *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 2, pp. 769-782, 1 March-April **2022**.
27. Z. Zhou, M. Shojafar, M. Alazab, J. Abawajy and F. Li, "AFED-EF: An Energy-Efficient VM Allocation Algorithm for IoT Applications in a Cloud Data Center," in *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 2, pp. 658-669, June **2021**.
28. Zhou Zhou, Jemal Abawajy, Morshed Chowdhury, Zhigang Hu, Keqin Li, Hongbing Cheng, Abdulhameed A. Alelaiwi, Fangmin Li, "Minimizing SLA violation and power consumption in Cloud data centers using adaptive energy-aware algorithms," *Future Generation Computer Systems*, Vol. 86, **2018**, Pages 836-850, ISSN 0167-739X.
29. Z. Zhou, M. Shojafar, M. Alazab and F. Li, "IECL: An Intelligent Energy Consumption Model for Cloud Manufacturing," in *IEEE Transactions on Industrial Informatics*, vol. 18, no. 12, pp. 8967-8976, Dec. **2022**.
30. C. Yu, S. Shen, H. Yang, K. Zhang and H. Zhao, "Leveraging Energy, Latency, and Robustness for Routing Path Selection in Internet of Battlefield Things," in *IEEE Internet of Things Journal*, vol. 9, no. 14, pp. 12601-12613, 15 July15, **2022**.
31. K. Nichols, S. Blake, F. Baker, and D. Black, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, IETF RFC 2474, Dec. **1998**.
32. J. Nagle, "On Packet Switches with Infinite Storage," in *IEEE Transactions on Communications*, vol. 35, no. 4, pp. 435-438, April **1987**.
33. ITU-T Recommendation, Network performance objectives for IP-based services, Y.1541, Feb. **2006**.