Evaluation of Middleware for Bandwidth Aggregation using Multiple Interface in Wireless Communication

Etsuko Miyazaki
Ochamonizu University
2-1-1 Ohtsuka, Bunkyo-ku 112-8610
Tokyo, Japan
(Currently with Microsoft Corp.)

Masato Oguchi
Ochamonizu University
2-1-1 Ohtsuka, Bunkyo-ku 112-8610
Tokyo, Japan
Email: oguchi@computer.org

***Abstract*—Although a variety of wireless interfaces are available on mobile devices, they still provide only low throughput so far. When coverage areas of those different technologies overlap, mobile devices with multiple interfaces can use them simultaneously by mechanism of Bandwidth Aggregation. However, there are some performance problems for Bandwidth Aggregation on Network Layer and lower Layer which derive from TCP congestion control mechanism. If Bandwidth Aggregation is performed at a layer lower than Transport layer, a packet loss happend in one route should decrease performance of all routes because reducing TCP congestion window on such a case affects communications of all routes. Thus we have proposed advanced Bandwidth Aggregation on Middleware for the purpose of avoiding there problems. If Bandwidth Aggregation is performed at Middleware that locates between Transport layer and applications, TCP congestion windows can be managed separately as route by route. In this paper, we have evaluated Middleware for Bandwidth Aggregation, which includes throughput and buffer size of receiver Middleware. According to the evaluation, it is possible to prevent from performance degradation when a packet loss happens using Middleware with an appropriate size of buffer at receiver-side.**

***Keywords*-component; Bandwidth Aggregation; Multiple Interface; Middleware; Buffer Size; IEEE 802.11; TCP Congestion Window**

I. Introduction

The growth of mobile Internet communication stimulate developments of a variety of wireless technologies: for example, IEEE 802.11, Bluetooth, and Worldwide Interoperability for Microwave Access (WiMAX). Although some of them have relatively broad bandwidth, they still have lower throughput than wired connection such as Ethernet, and are able to be accessed only in limited areas. It is possible to realize more efficient mobile Internet service using multiple interfaces simultaneously, when we are in areas covered by several services of wireless technologies. Bandwidth Aggregation which uses multiple interface simultaneously is proposed as advanced way to access Internet from mobile node.

Among several research works, seamless vertical handoff from one interface to another has been addressed [2]. One of the advanced form of this technology is known as cognitive radio. In such a system, it is possible to change a wireless connection from one radio wave frequency band to another, depending on the condition of radio wave. There are two types in cognitive radio; one is shared frequency type in which available frequency is chosen and used dynamically, and the other is heterogeneous type in which different kinds of wireless systems such as IEEE802.11 WiFi, WiMAX, and Long Term Evolution (LTE) have been chosen and used dynamically. These technologies have already been practical and expected to begin its commercial service in the near future. For example, NEC Corp. has demonstrated that, based on OpenFlow technology [3], it is possible to change a connection from WiFi to WiMAX dynamically based on the load of each connection [4].

Although we are able to change from one wireless connection to another, we have not achieved Bandwidth Aggregation in practical use. This is a little more complicated than vertical handoff, in which cognitive radio technologies and aggregation technologies should be considered simultaneously. There are not only implementation difficulties but also performance matter for aggregating multiple connections. If Bandwidth Aggregation is realized in a mobile environment, this gives us better mobility support, reliability and resource sharing.

Thus, we have proposed and evaluated an innovative mechanism of Bandwidth Aggregation in this paper. While we have focused on aggregation of several WiFi connections, it is possible to apply this proposal and the evaluation results to heterogeneous aggregation of WiFi, WiMAX, and LTE, for example.

The rest of paper is organized as follows. First, background of Bandwidth Aggregation is discussed in Section II. In Section III, our proposed model for Bandwidth Aggregation is introduced. An outline of evaluation of the proposed method is mentioned in Section IV. Various scenarios of experiments are introduced and buffer size of receiver Middleware is evaluated in Section V. In Section VI, the method of Bandwidth Aggregation on Network layer and the proposed methods are compared. Finally, concluding remarks are related in Section VII.

II. Background of this research work

A. Bandwidth Aggregation in Various Layers

Bandwidth Aggregation is supposed to be realized on several layers, while they have merits and demerits respectively.

An approach on Datalink layer [5] will give the most effective result, and upper layers do not need to care about Bandwidth Aggregation. However, we can install it only world using same protocol for datalink layer and have to install specific hardware to their nodes. In other words, all network interfaces should be replaced to use this approach.

An implementation in Network layer will provide efficient Bandwidth Aggregation by intelligent methods [6][7]. The advantages using Network layer are they perform transparently to widely used Transport protocol such as TCP and UDP. However, TCP may not achieve estimated efficiency due to a possibility that they receive packets in incorrect order. Although only the incorrect order might not necessarily be a problem, this causes congestion control more than required. That is to say, this may cause unnecessary packet retransmission and reduce performance of all connections [8]. Although Reordering-Robust TCP (RR-TCP) is proposed to prevent this phenomenon [9], it is impossible to recover it when the volume of packet loss is larger than a certain level.

In Transport layer, they have congestion window for each path. It enables more effective transport by doing packet distribution and retransmission for each path [10]. However, the system has to be installed into each operation system in all the end-end way.

An implementation on Application layer does not demand to replace current operating systems [11]. However, there are variety of applications and it is difficult to implement aggregation method for all of them. After connections established, we have to consider how to distribute packets for each connection.

B. Packet Loss Problem in Bandwitdth Aggregation on Network Layer

If multiple interfaces are used for concurrent communications, there are possibilities that receiving node may take packets incorrect order. In such a case, receiver recognizes occurring of packet loss incorrectly due to receiving packets different from expected order of packets. Then TCP requests retransmission unnecessarily. This is one of problems in Bandwidth Aggregation on Network Layer. Although incorrect judgement of packet loss is not only the problem of Bandwidth Aggregation, the problem becomes complicated because multiple packets are delivered through different routes.

For the purpose of eliminating this problem, Earliest Delivery Path First (EDPF) was proposed [6]. EDPF is implemented to the node that delivers packets to different paths. EDPF chooses on which path each packet should be sent in consideration of their bandwidth, delay and congestion. EDPF decides the fastest path to transmit the packet to receiver node. All packets are sent through the route on which estimated time is the shortest. Therefore, receiver can receive any packets in correct order. It makes Bandwidth Aggregation effective as estimated efficiency in no packet loss circumstances, and its effectiveness has been verified by previous researches.

C. Performance Problem in Bandwidth Aggregation on Network Layer

In the case of wireless communication, there are so many packet losses more than the case of wired communication. When Bandwidth Aggregation is operating on Network layer or lower layer, TCP cannot recognize which path causes the packet loss. Thus, TCP executes congestion control and throughput is degraded more than necessary. This is the second problem in Bandwidth Aggregation on Network Layer.

Packet-Pair based Earliest-Delivery-Path-First algorithm for TCP applications (PET) and Buffer Management Policy (BMP) were proposed for the purpose of fixing that problem on Network layer [8]. PET has functions estimating which path should be used more strictly and dynamically. BMP is implemented in receiver node, evaluates whether a received packet is needed to line up or caused packet loss. When BMP receives later sequence number packet, it informs packet loss was occurred for sure. Otherwise BMP delivers correct order packets to TCP.

With PET and BMP, more effective communication is realized compared with implemented EDPF, in particular, when packet losses occur. However, in circumstances with a lot of packet losses, even PET-BMP cannot execute efficient Bandwidth Aggregation. This is because TCP congestion window is reduced when a packet loss is considered to happen at one connection. As a result, throughput of all connections is degraded since TCP congestion window is shared among all connections in the case of Bandwidth Aggregation on Network layer.

This is one of the most difficult problems to solve in Bandwidth Aggregation on Network Layer. Referenced researches claim that it is possible to achieve expected results with eliminating packet losses using other methods [8]. In reality, it is too difficult to eliminate packet losses in wireless communication.

III. Our Proposal for Bandwidth Aggregation

As shown in previous chapters, we face various obstacles using Bandwidth Aggregation on Network layer and/or lower layer. Thus, we have proposed Middleware layer that aggregate bandwidth on the middle between Application layer and Transport layer. Figure 1 shows comparison between Bandwidth Aggregation on Network layer and our proposed model. In our model, all TCP connections are treated separately and aggregated them at the Middleware.

A. An Overview of Our Proposal

Our proposed model has different TCP connections per paths and aggregates their connections at the Middleware. Therefore, applications are not required to be conscious of aggregating bandwidth. It uses independent TCP congestion windows per paths, which prevent throughput degradation more than necessary, explained in the previous section, in the case of many packet losses.

This feature avoids the problems that happen in implementation on Network layer. If the bandwidth is aggregated on Network layer, TCP cannot determine on which path packets are lost, because TCP is upper layer and receives only after data is aggregated. Our previous research work shows their problems on Network layer are solved [12]. The defect which PET-BMP could not solve is overcome by our method, which means Bandwidth Aggregation on Middleware is more effective than that on the other layers.

This approach can also be implemented by modifying TCP which aggregates some connections on Transport layer. However, with the easier way with Middleware, we can use existing TCP for the purpose of achieving the most efficient Bandwidth Aggregation.

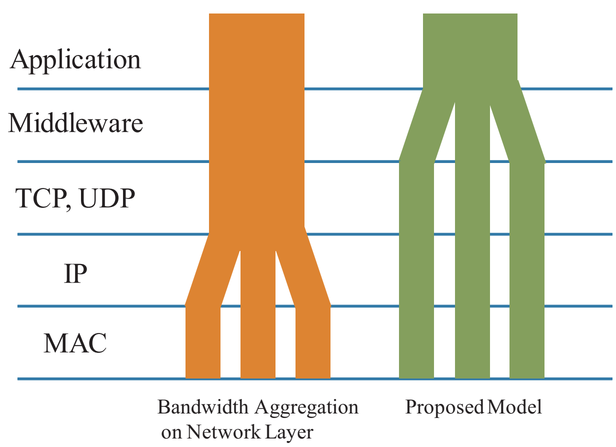

Figure 1. Comparison Between Bandwidth Aggregation on Network Layer and Our Proposed Model

B. The Design of Our Proposed Model

The sender Middleware establishes TCP connections on all possible paths. Packet of sent data is handed over from an application to Middleware, and a sequence number is given to a packet. A packet is sent out through enabled connection. EDPF for Middleware (Earliest Delivery Path First for Middleware) is used as an algorithm for the route selection when the packet is sent out.

When EDPF is used for the packet delivery, it calculates how long it takes from the sender to a receiver on each route, and it chooses the fastest path. For the estimation of the packet delivery, the bandwidth of wired and wireless route, delay time, and the congestion state is used. For the sake of next estimation, these parameters are updated with every packet delivery.

On the other hand, the receiver Middleware should put received packets in correct order and give them to an appropriate application. The receiver Middleware has a possibility that some packets arrive in incorrect order and needs to have a buffer to keep packets, for the purpose of waiting for the packet with expected sequence number. Thus, Middleware has a role to reorder the packets and hands over to the application afterward.

Estimation of required buffer size in each circumstances is one of the significant points for designing the Middleware. BMP also discusses about buffer size and controls how packets should be deriverd. We propose the method on other layer and suppose that they will behave differently.

IV. Evaluation with Simulation Software

In this experiments, we are motivated by the advantages that uses Bandwidth Aggregation through simultaneous use of multiple interfaces. We have used simulation software QualNet for the experiments [13].

For the purpose of designing Middleware, the buffer size of Middleware receiver has to be estimated clearly. We have investigated the size under various circumstances in Section V. The ratio of bandwidths between multiple wireless connection is changed and the required buffer size is evaluated at each case.

In Section VI, we have evaluated our proposed method. Since we have used EDPF for Middleware as an algorithm for routing in Bandwidth Aggregation, we have compared this method with a simple Weighted Round Robin (WRR) algorithm. Bandwidth Aggregation at Network layer, which is evaluated in existing literature, has been implemented and evaluated at first. That is to say, the existing method has been double-checked by an experiment. Next, our proposed method, Bandwidth Aggregation at Middleware, has been implemented and compared with the existing method in terms of performance stability when a packet loss occurs.

V. Evaluation of a Buffer Size of Receiver Middleware

In order to design Bandwidth Aggregation on Middleware, one of significant parameter value is a required buffer size of receiver Middleware. Therefore, we have evaluated the buffer size in various cases by changing the number of available routes, bandwidth, and delay time. With this experiment, we have determined the required buffer size of receiver Middleware.

For the evaluation, a simple WRR is used as a packet delivery algorithm, In WRR, a packet delivery route is determined depending on the ratio of wireless part bandwidths. For example, if the bandwiths of wireless part of each route are 200kbps, 100kbps, and 50kbps, respectively, the ratio of packet delivery to each route is 4:2:1.

A. Low Bit Rate Wireless Communications

As Scenario 1, the case with low bit rate wireless communications is evaluated. This is shown in Figure 2. Node 1 sends data to Node2, which has 2 wireless interfaces and receives data through 2 paths.

The bandwidth at wired connection is 10Mbps in this scenario. One of wireless connections is fixed to 100kbps and the other is varied from 100kbps to 800kbps. That is to say, the ratio of two bandwidths of wireless connections is varied from 1:1 to 1:8. Transport protocol used in this evaluation is TCP new Reno, and parameters are configured as Table I.

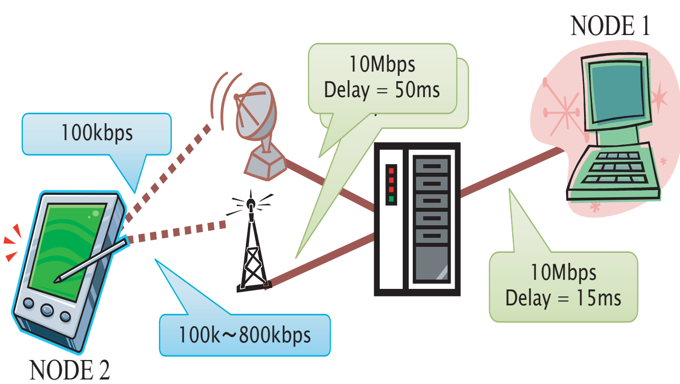

Figure 2. An Overview of Scenario 1

Table I
TCP PARAMETERS

MSS	1,460Bytes
Send buffer	65,535Bytes
Receive buffer	65,535Bytes

B. High Bit Rate Wireless Communications

As Scenario 2, the case with relatively high bit rate wireless communications is evaluated. This is shown in Figure 3. Node 1 sends data to Node2, which has 2 wireless interfaces and received data through 2 paths. The bandwidths of wireless connections as well as wired connections are different from the case of Scenario 1.

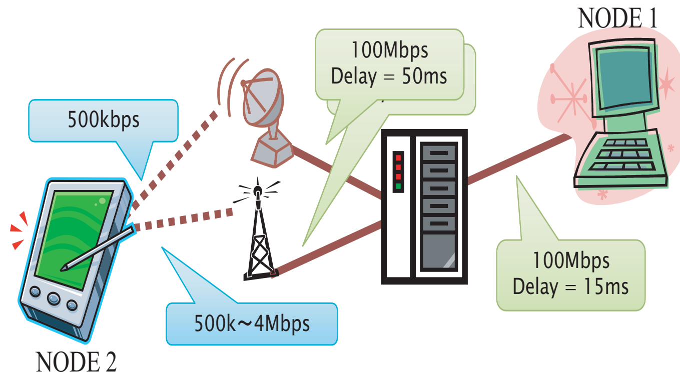

Figure 3. An Overview of Scenario 2

The bandwidth at wired connection is 100Mbps in this scenario. One of wireless connections is fixed to 500kbps and the other one is varied from 500kbps to 4Mbps. The ratio of two bandwidth of wireless connection is varied from 1:1 to 1:8, which is the same with the case of Scenario 1. Transport layer protocol and TCP parameters are also configured as the same with Scenario 1.

C. Evaluation in Various Cases

We have also evaluated two more other cases. In Scenario 3, as shown in Figure 4, while the bandwidths of wireless connections are the same with that of Scenario 2, that is, one of wireless connections is fixed to 500kbps and the other one is varied from 500kbps to 4Mbps, the bandwidth of wired connection is 50Mbps, the half size of that in Scenario 2.

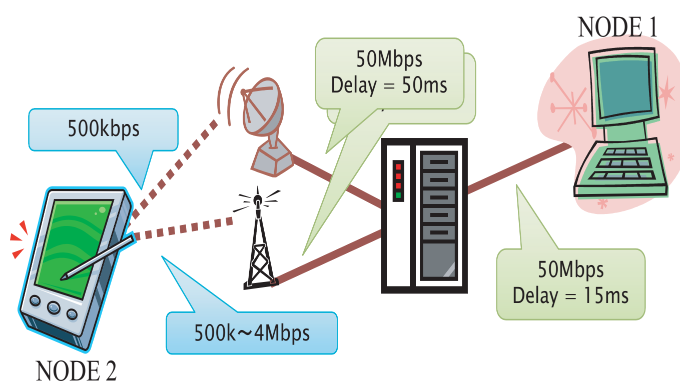

Figure 4. An Overview of Scenario 3

In Scenario 4, as shown in Figure 5, the bandwidth of wired connection is 50Mbps, which is the same with that of Scenario 3. However, the bandwidths of wireless connections are the half size of those in Scenario 3, that is, one of wireless connections is fixed to 250kbps and the other one is varied from 250kbps to 2Mbps.

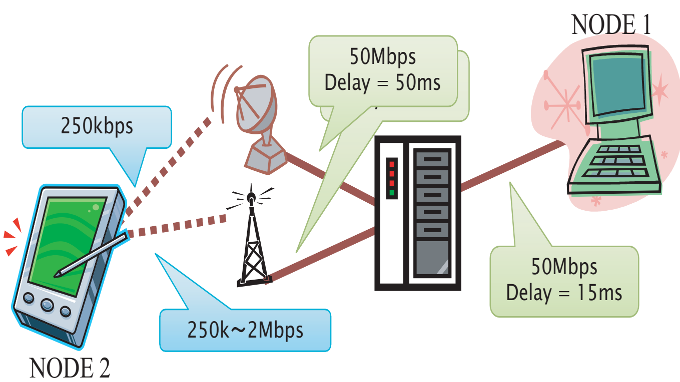

Figure 5. An Overview of Scenario 4

In both scenarios, the ratio of two bandwidth of wireless connection is varied from 1:1 to 1:8, which is the same with the previous scenarios. Transport layer protocol and TCP parameters are also configured as the same with previous Scenarios.

We have evaluated the required buffer size of receiver Middleware using these models. The experimental results are shown in the following subsections.

D. Period of Steady State and Unsteady State

Figure 6 shows throughputs of two connections and buffer size of receiver Middleware when bandwidths of wireless connections are set to 100kbps and 300kbps in Scenario 1.

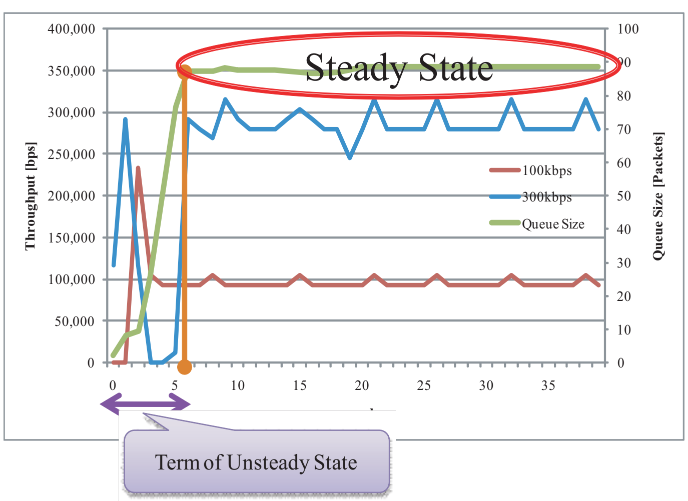

Figure 6. Throughputs and Queue Size

At beginning, the communication is a little unstable for a while. After a short period, two of wireless connections' throughput show stable and efficient communication.

Queue size of receiver Middleware is growing at first and becomes stable at a value. We call the period that buffer size is stable "Steady State", and the time until being stable "Term of Unsteady State". We focus on their values at various circumstances.

E. Association Between Ratio of Bandwidths and Required Buffer Size

Figure 7 shows required buffer size for Middleware of reciever at the period of Steady State in Scenario 1 when ratio of two bandwidths is changed from 1:1 to 1:8.

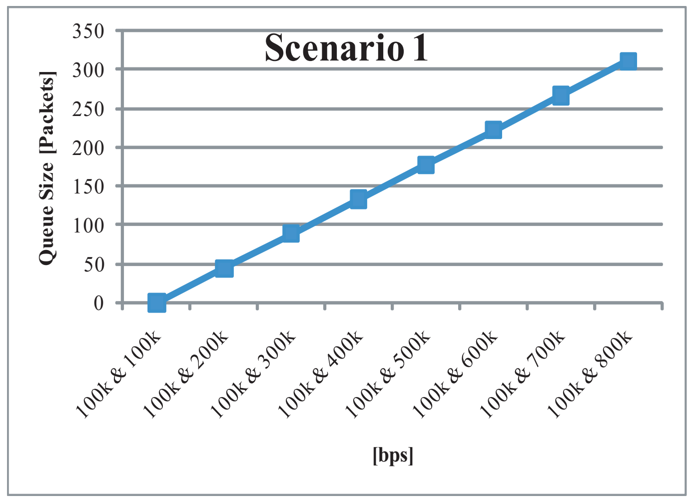

Figure 7. Buffer Size in Scenario 1

The value of buffer size when two interfaces have the same bandwidths is zero. That is to say, almost no buffer is required when two wireless connections have the same bandwidth.

On the other hand, when two interfaces have different bandwidths, the required buffer size is proportional to the ratio of one interface's bandwidth to the other.

Figure 8 shows the required buffer size at the period of Steady State in Scenario 2 when ratio of two bandwidths is changed.

The value of required buffer size is proportional to the ratio of one interface's bandwidth to the other, as is the same with Scenario 1. Although Scenario 1 and Scenario 2 have different bandwidth and different ratio of bandwidth between wired and wireless connections, buffer size is only

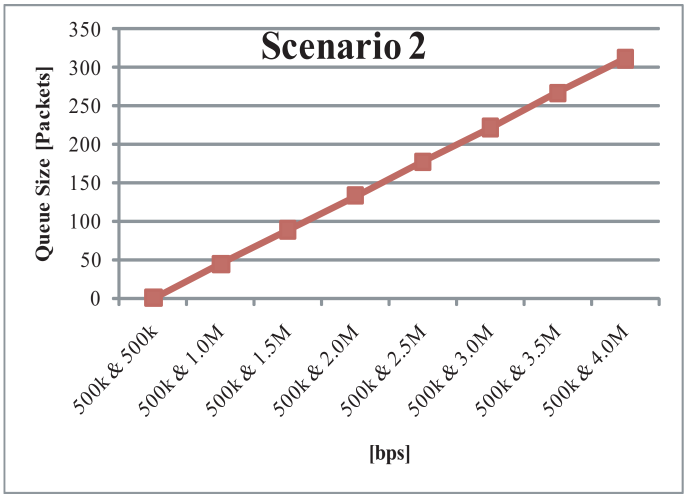

Figure 8. Buffer Size in Scenario 2

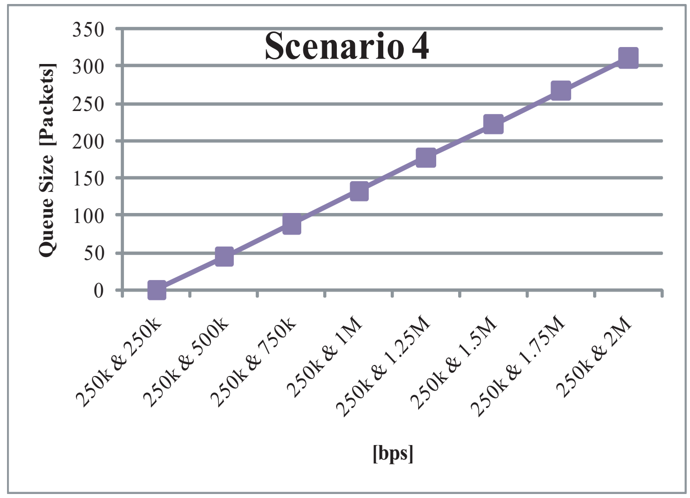

Figure 10. Buffer Size in Scenario 4

determined by the ratio of two bandwidths of wireless connections.

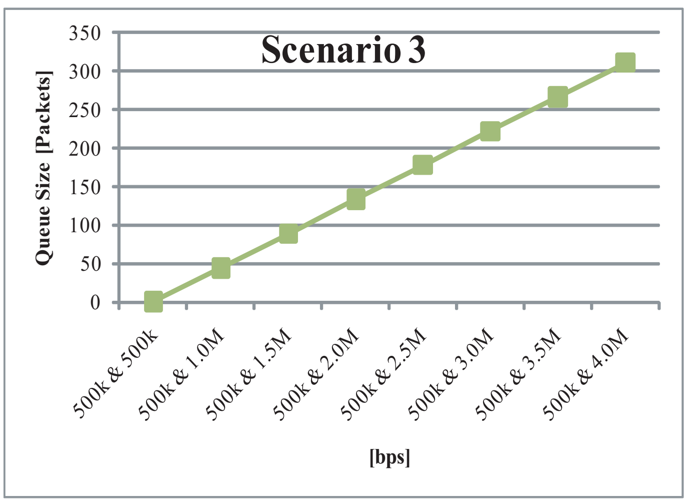

Figure 9. Buffer Size in Scenario 3

In addition, the required buffer sizes at the period of Steady State in Scenario 3 and 4 are shown in Figure 9 and 10, respectively. In these figures also, the value of required buffer size is proportional to the ratio of one interfaces' bandwidth to the other.

In all scenarios, the absolute value of the buffer size is completely the same when the ratio of two interfaces' bandwidths is the same. This is one of the most significant parameters to design Middleware, and it is possible to determine the value like this. It is interesting to see that required buffer size can be determined only by the ratio of bandwidths of two wireless interfaces, regardless of their absolute values.

F. Association Between Ratio of Bandwidths and Period of Unsteady State

Figure 11 shows period of Unsteady State, the period until throughput becomes stable from the beginning, when the ratio of two wireless connections' bandwidths is changed. All cases of Scenario 1 to 4 are shown in this figure.

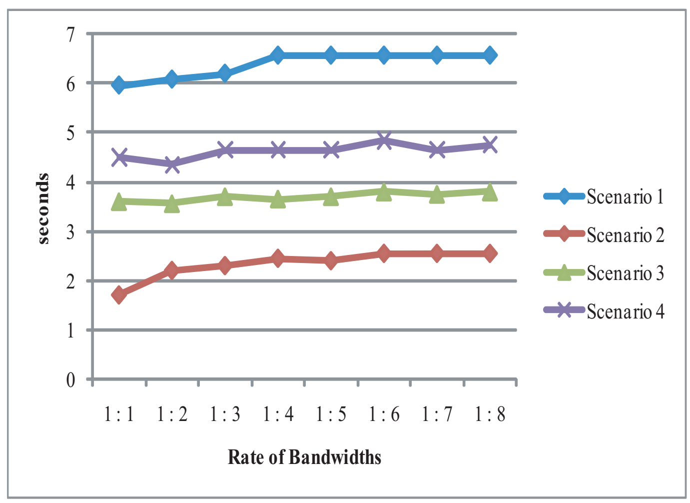

Figure 11. Period of Unsteady State

The period of Unsteady State in Scenario 2, whose connections have higher bit rate, is shorter than that of Scenario 1, whose connections are lower bit rate, in all cases. The ratio of bandwidths between two wireless connections does not affect their length.

VI. Comparison between Bandwidth Aggregation on Network Layer and Our Proposed Method

In this section, Bandwidth Aggregation on Network layer and that on Middleware are evaluated. In order to evaluate the effectiveness of EDPF for Middleware as an algorithm for routing used in Bandwidth Aggregation, this is compared with the simpler method, Weighted Round Robin (WRR).

First, the evaluation result of Bandwidth Aggregation on Network layer has been double-checked by an experiment. Next, performance of Bandwidth Aggregation on Middleware, EDPF for Middleware proposed in this paper, has been evaluated and compared with that of Bandwidth Aggregation on Network layer.

A. An Overview of Experiment

In this evaluation, as shown in Figure 12, a mobile terminal that has three wireless connections (Node 2) receives data sent from Node 1, using three routes simultaneously.

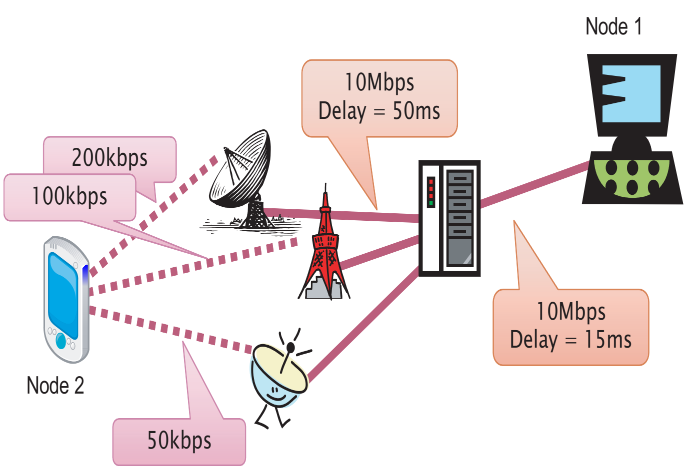

Figure 12. Scenario of Experiment

The bandwidth of wired connection, presented in a real line, is 10Mbps and the delay time is presented in the figure. The bandwidths of wireless connections are 200kbps, 100kbps, and 50kbps, respectively. Transport layer protocol used in this experiment is TCP new Reno, and TCP parameters are set as shown in Table II.

Table II
TCP Parameter

MSS	1,460Bytes
Send buffer	65,535Bytes
Receive buffer	65,535Bytes

B. Experiment with No Packet Loss

For Bandwidth Aggregation on Network layer and on Middleware, EDPF and WRR are used as routing algorithm for data delivery. In the first case, throughput and required buffer size have been evaluated with no packet loss environment in Figure 12.

1) Evaluation Result of Bandwidth Aggregation on Network Layer: Throughput when two routing algorithms are used, EDPF and WRR, with Bandwidth Aggregation on Network layer is compared in Table III. In this table, the number of Duplicated ACK and the number of retransmission in both cases are also indicated.

Table III
Experimental Result

Algorithm	Thr(kbps)	Dup ACKs	Retransmitted
EDPF	329	0	0
WRR	265	522	96

In this experiment, the sum of bandwidths of three routes is 350kbps. Therefore, while performance of WRR is 75.7% of total bandwidth, 94.0% in the case of EDPF.

This is because no Duplicated ACK occurs and no retransmission of packet is observed in the case of EDPF as shown in Table III. Thus the performance of EDPF is higher.

2) Evaluation Result of Bandwidth Aggregation on Middleware: Throughput when EDPF for Middleware is used for Bandwidth Aggregation on Middleware is shown in Figure 13. Throughput of each route is shown in this figure.

For the comparison, throughput when WRR is used for Bandwidth Aggregation on Middleware is shown in Figure 14.

Compared with the simpler WRR, EDPF for Middleware, which delievers data based on the situation of each route, achieves stable communication at each moment. That is to say, EDPF for Middleware has succeeded in stabilizing the behavior of multiple connections.

The volume of data buffered in receiver Middleware, using EDPF and WRR, is shown in Figure 15.

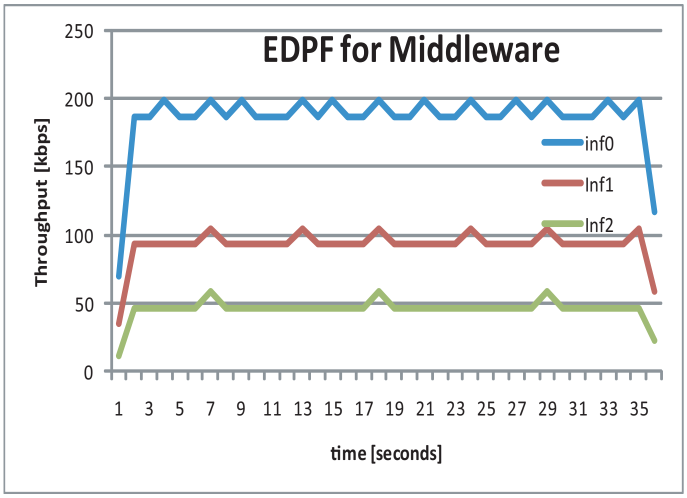

Figure 13. Throughput of EDPF for Middleware

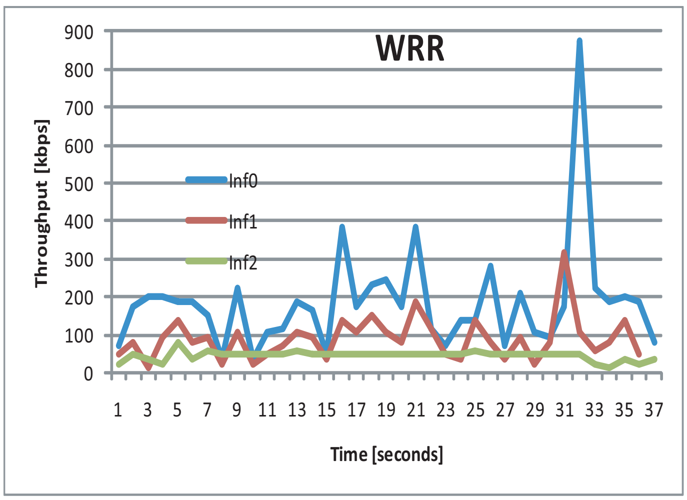

Figure 14. Throughput of WRR

In the case of using EDPF, the maximum number of packets stored in the buffer of receiver Middleware is only three, while the maximum number of packets is about 100 when WRR is used. Therefore, throughput of WRR is not stable as shown in Figure 14, and delay becomes large from a sender application to a receiver as a result.

C. Experiment with Packet Loss

In Bandwidth Aggregation on Network layer and that on Middleware, using EDPF as an algorithm of routing, the case with one packet loss 18 seconds after the beginning of the scenario is evaluated by observing throughput at each route.

1) Evaluation Result of Bandwidth Aggregation on Network Layer: Throughput of Bandwidth Aggregation on Network layer with packet loss is shown in Figure 16. The

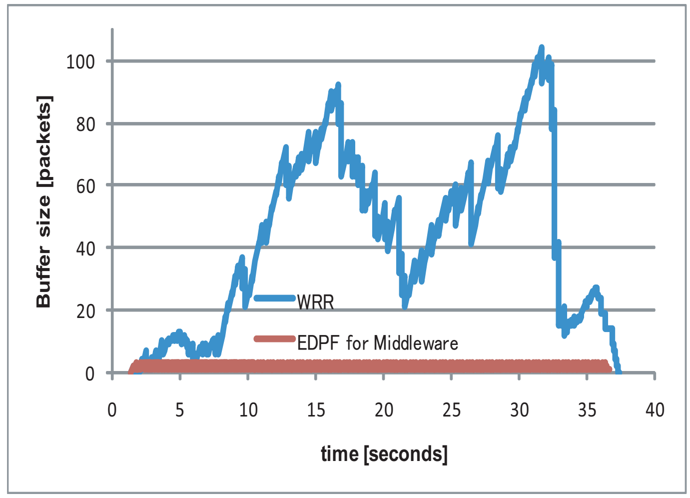

Figure 15. Buffer Size of Receiver Middleware

packet loss has occurred on one of three connections.

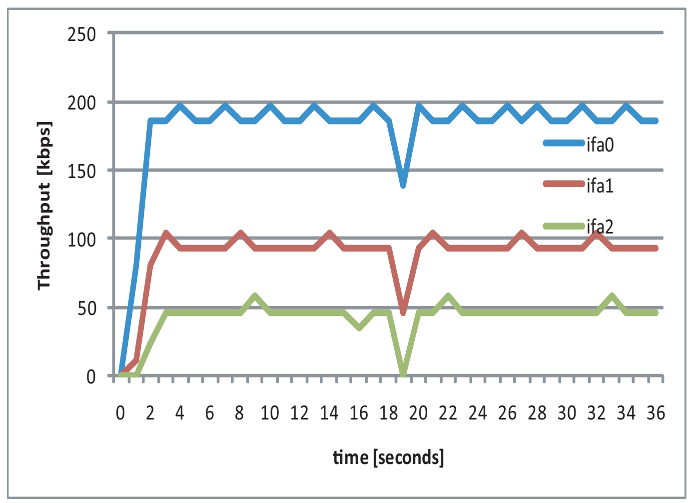

Figure 16. Throughput of Bandwidth Aggregation on Network Layer with Packet Loss

It is observed that throughput of all interfaces is reduced when the packet loss occurs, which is 18 seconds after the beginning of this scenario. In this case, packet loss at one connection affects other connections, and thus, total throughput is reduced. This is because TCP cannot decide on which connection the packet loss occurs, and congestion window of all connections should be decreased as a result. According to this evaluation results, it is impossible to achieve high performance with Bandwidth Aggregation on Network layer in the case of packet loss.

2) Evaluation Result of Bandwidth Aggregation on Middleware: Figure 17 shows the throughput of Bandwidth Aggregation on Middleware, using EDPF algorithm for packet routing, with packet loss on one of three connections.

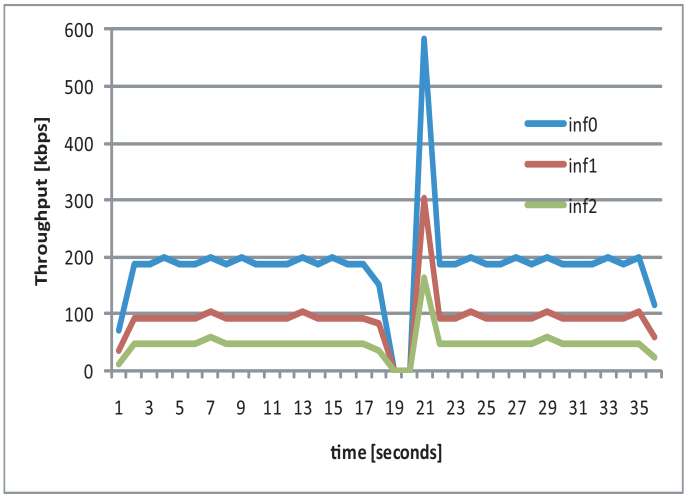

Figure 17. Throughput of Bandwidth Aggregation on Middleware with Packet Loss

In this case, the number of packet stored at buffer of receiver Middleware is shown Figure 18.

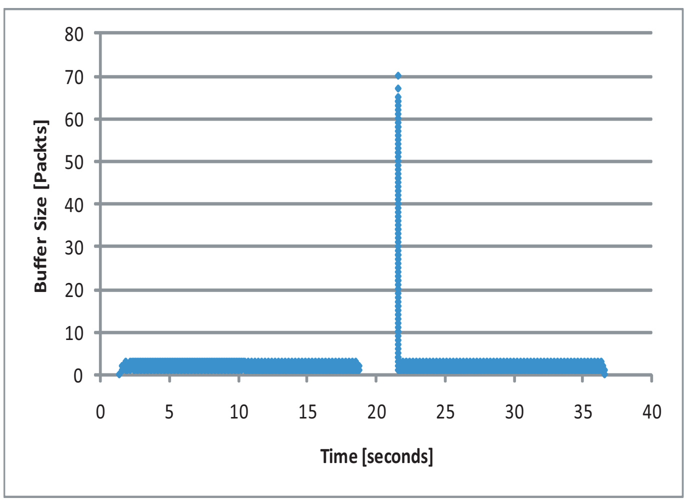

Figure 18. Buffer Size of receiver middleware

According to Figure 17, it seems that a single packet loss happened in one route reduces throughput of all routes. This seems something similar with the case of Bandwidth Aggregation on Network layer.

However, this figure shows the volume of data that can be passed from Middleware to an application. As shown in Figure 18, the volume of data stored in the buffer of receiver Middleware becomes large in order to wait for a retransmission of the lost packet. This causes performance degradation of all routes only temporarily and it is recovered afterwards as shown in Figure 17. That is to say, this is different from the phenomenon of throughput reduction in the case of Bandwidth Aggregation on Network layer. In this case, congestion windows are not decreased and data transfer continues during that period, and the data transfered during that period is stored in the buffer.

VII. Conclusion and Future Work

In this paper, we have experimented with network simulator for the purpose of evaluation of the communication using multiple interfaces simultaneously. The methods of Bandwidth Aggregation on network layer still have problems, for instance, because they can not recognize which path causes the packet loss. We have proposed the model of Bandwidth Aggregation on Middleware in order to eliminate the problem. The effect are verified compared with previous method since we can get comparable throughput as well as aggregating throughput of multiple connection.

The receiver Middleware needs to have buffer to restore the order of packets' sequence number. We have investigated how large buffer is needed in various situations. The mobile node which has two interfaces varies one of interface's bandwidth and observes the buffer size. The result shows it proportional to the ratio of one interface's bandwidth to other one.

As a routing algorithm for Bandwidth Aggregation, EDPF for Middleware and simple WRR are compared. As a result, EDPF for Middleware achieves stable and efficient performance of communications. In addition, experiment of a case with packet loss is performed. According to the evaluation results, Bandwidth Aggregation on Middleware is able to perform superior communications compared with Bandwidth Aggregation on Network layer.

In the future, we will implement the feature of buffer size that demonstrated by the experiments and function on the sender Middleware considering how to distribute each packets to the paths. Moreover, we will suppose that mobile node can have many wireless interfaces and study the result in such cases. In addition, we try to achieve more efficient Bandwidth Aggregation in a various situations, for instance, various pattern of lower layer and dynamically-changed bandwidth.

Acknowledgement

We would like to thank to Dr. Onur Altintas in TOYOTA InfoTechnology Center, Co. for the conscientious advice and help for this research work.

REFERENCES

[1] E. Miyazaki and M. Oguchi : "Evaluation of Buffer Size for Middleware using Multiple Interfaces in Wireless Communication," Tenth International Conference on Networks (ICN2011), pp.202-205, January 2011.

[2] M. Stemm and R. Katz : "Vertical handoffs in wireless overlay networks," Mobile Networks and Applications Vol.3, No4, pp.335-350, January 1998.

[3] OpenFlow : http://www.openflowswitch.org/

[4] NEC Corp. : http://www.nec.co.jp/press/ja/1002/0402.html, February 2010.

[5] IEEE P802.3ad Link Aggregation Task Force : http://grouper.ieee.org/groups/802/3/ad/

[6] K. Chebrolu and B. Raman : "Bandwidth Aggregation for Real-Time Applications in Heterogeneous Wireless Networks," IEEE Transactions on Mobile Computing, Vol.5, No4, pp.388-403, April 2006.

[7] K. Koyama, Y Ito, H. MINENO and S. Ishihara: "Evaluation of Performance of TCP on Mobile IP SHAKE," Transactions of Information Processing Society of Japan, 2004.

[8] K. Chebrolu, B. Raman, and R.R. Rao : "A Network Layer Approach to Enable TCP over Multiple Interfaces," Journal of Wireless Networks (WINET), Vol.11, No5, pp.637-650, September 2005.

[9] M. Zhang, B. Karp, S. Floyd and L. Peterson: "RR-TCP: A Reordering-Robust TCP with DSACK," IEEE International Conference on Network Protocols, 2003.

[10] M. Zhang, J. Lai, A. Krishnamurthy, L. Peterson and R. Wang: "A transport layer approach for improving end-to-end performance and robustness using redundant paths," USENIX 2004 Annual Technical Conference, pages 99-112, 2004.

[11] H. Nozawa, N. Honda, K. Sakakibara, J. Nakazawa, and H. Tokuda: ARMS: "Application-level Concurrent Multipath Utilization on Reliable Communication," Internet Conference 2008, October 2008.

[12] E. Miyazaki, O. Altintas, and M. Oguchi : "A Study of Bandwidth Aggregation Using Multiple Interfaces on Middleware Layer," DICOMO 2010, July 2010.

[13] Scalable Network Technologies : http://www.scalable-networks.com/