A Path Switching Scheme for SCTP Based on Round Trip Delay

Fang-Yie Leu  
Tunghai University  
Department of Computer Science  
Taichung, Taiwan  
leufy@thu.edu.tw

Yong-Lon Chen  
Tunghai University  
Department of Computer Science  
Taichung, Taiwan  
g96350030@thu.edu.tw

Abstract—In this paper, we propose an aggressive path switching scheme for SCTP. Before data transmission, the scheme selects the fastest path as the primary path to transmit packets. When the path fails or transmission quality is poor, this scheme evaluates all alternate paths, and selects the one with the best quality as the new primary path to substitute for the original one. After that, packets are delivered through the new path. Several factors are considered in the evaluation, including bandwidth, encryption/decryption, size of the congestion window, retransmission policy, routing policy, etc.

Keywords: SCTP, congestion control, retransmission, routing, bandwidth

I. INTRODUCTION

In recent years, stream control transmission protocol (SCTP) has been quickly developed and widely deployed. Its importance in wireless communication is greater every day. Leu [1] employed SCTP as a key mechanism of network mobility to achieve a seamless handover, particularly for delivering multimedia data packets. Many modified versions, like CS-SCTP [2], and nSCTP [3], have been proposed. However, most SCTP systems consider only a few performance-affecting factors, implying that their performance can be further improved.

In this study, we develop a new path switching scheme for SCTP, called the path selection and switching process (PSASP), which chooses the best path for SCTP by evaluating mechanisms and activities that influence SCTP transmission efficiency, including size of encrypted/decrypted data [2], size of congestion window, retransmission policies, length of a routing path, a packet’s round trip time (RTT) [4], network delays, hardware speed and bandwidth, etc. These mechanisms and activates are dispersed in layers of the OSI model. For example, routing is a layer-three task, and hardware speed is a layer one concern. We also propose a path switching scheme based on evaluation of the results of the related mechanisms and activities, which can help the SCTP to select the best primary path. Experimental results show that this scheme can effectively improve transmission performance, and decrease packet loss rates and jitters.

This article is organized as follows. Section 2 introduces relevant background and related work. Section 3 describes our system architecture. The experimental results are presented in section 4. Section 5 concludes this article and addresses our feature work.

II. BACKGROUND AND RELATED WORK

2.1 SCTP

The SCTP inherits features and attributes from TCP, but provides new features for users [4], including multi-homing, multi-streaming, heartbeat, four-way handshake, and chunk bundling. Among them, the former two are the most important ones.

1) Multi-homing: An SCTP association often contains multiple paths, each of which is an ip-to-ip connection. When transmission quality is poor, it chooses one of the secondary paths, known as alternate paths, to substitute for the primary path.

2) Multi-streaming: this divides a path into multiple subpaths, called streams. All streams are independent of each other in transmission. Before data transmission, SCTP defines a number of streams and assigns packets to streams for transmission to prevent the head of line problem.

2.2 Related Work and Influential Factors

According to previous studies [1-6], network transmission is influenced by several factors. Yang, Chang and Huang [2] mentioned that encryption, due to requiring additional overheads, makes a data chunk include much more information than transmitting plain text does. The overheads consume extra packet processing time and transmission time.

Dahel and Saikia [4] stated that round-trip time (RTT) of a path responds to the current available bandwidth of the path, and can help to determine when to switch data transmission to a good path. The RTT based congestion avoidance (RBCA) scheme and Switch Path on Congestion (SPC) are then proposed to calculate the RTT based on the SACK timestamp option, and then the SCTP refers to the RTT to adjust cwnd and select a new primary path. Al-kaisan et al. [5] presented a modified version of the SCTP, called the optimized SCTP, with which cwnd reduction of packet loss is slow, i.e., cwnd = cwnd − [0.05 * cwnd] rather than reducing cwnd by half.

Further, the standard SCTP does not clearly define how to select one of the alternate paths as the primary path. In fact, the round-trip time is a good method to control path switching. But
round-trip delay is a complicated delay consisting of several path performance affecting factors which are dispersed among different network layers. In this study, we will analyze how the factors affect path performance. Based on the analysis, we can then select the best alternate path as the new primary path when the performance of the original primary path is poor.

III. THE PROPOSED SCHEME

In a multi-homing environment, the PSASP has two main steps in selecting the best path for an SCTP association. Step 1 is selecting an initial primary path. Before data transmission, the PSASP first checks to see whether any network flow flows through the path with the widest bandwidth or not. If not, the path will be selected as the initial primary path. Otherwise, the PSASP enters step 2 which evaluates performance for all paths of concern. In addition, when transmission fails or communication quality is poor, the PSASP will also invoke the step-2 process. In this process, dynamic influential factors, including length of encrypted/decrypted data, size of congestion window, current available bandwidth, etc. are involved to compute the packet delivery delay of a transmission path. The one with the minimum delay will be selected as the initial or the new primary path. In the following, we assume the initial bandwidth of an association and initial bandwidth of each path involved are known, where initial bandwidth = current available bandwidth + occupied bandwidth, and the packet arrival rate of each path segment along a path, e.g., the segment between nodes i and i+1, has a Poisson distribution.

3.1 Dynamic Factors

The following mechanisms and activities, including encryption/decryption, size of congestion window, and round-trip delay, are considered as key dynamic factors in selecting a primary path.

3.1.1 Encryption/Decryption

According to [2], an encrypted packet has a longer transmission delay than its original packet has because of additional processing efforts, such as packet encryption and decryption, and additional transmission overheads. Generally, encrypted packets can be classified into four security levels [2], and often higher security levels have more overheads.

3.1.2 Size of congestion window

When the size of a congestion window is relatively larger, implying the transmission path has better quality, and available bandwidth, i.e., initial bandwidth – occupied bandwidth, is wider, then a sender can transmit more data per second to its receiver. When the window size is small, it often means the available bandwidth of the path is limited, and the network quality is not good. Once packets are lost, the window size will be reduced to mitigate data flow, and to shorten the packet waiting queue. In this case, very often available bandwidth < initial bandwidth – traffic-occupied bandwidth. But in this study, we assume that available bandwidth = initial bandwidth – traffic-occupied bandwidth – SCTP_occupied bandwidth, where SCTP-occupied bandwidth results from shrunken congestion window size. If packets can be successfully and smoothly delivered to the destination, the window size will be slowly enlarged, which is known as a slow start.

3.1.3 Round Trip Delay

The RTT more accurately reflects real network speeds. Many systems employ it as an important performance parameter. In this study, we consider round trip delay as the key performance-measure parameters. The delay can be further divided into transmission, propagation, processing and queueing delays.

1) Processing delay: Processing delay is the time required to prepare and receive a packet, and encrypt and decrypt chunks. The purpose of these activities is basically getting the data ready to be transmitted or processed. Performance of the activities is mainly influenced by hardware speed. The items of concern are packet size, size of encrypted/decrypted data, and node’s data generating speed, encryption speed, receiving speed, decryption speed, and processing speed, which are described in Table I. Generally, three types of nodes are involved in deriving processing delays, including source node, intermediate nodes i.e. routers and switches, and destination node.

<table>
<thead>
<tr>
<th>Table I. The Speeds Involved in Packet Processing Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Items</strong></td>
</tr>
<tr>
<td>generating speed</td>
</tr>
<tr>
<td>encryption speed</td>
</tr>
<tr>
<td>receiving speed</td>
</tr>
<tr>
<td>decryption speed</td>
</tr>
<tr>
<td>processing speed</td>
</tr>
</tbody>
</table>

A. Cost for processing a data packet

a) For a source node Src, the time required to process a data packet Q, denoted by \( T_{d-s} (Src) \), is

\[
T_{d-s} (Src) = \frac{\text{size} (Q)}{\text{generating speed} (Src)} \cdot \frac{\text{size of encrypted data}}{\text{encryption speed} (Src)}
\]

Let \( Q' = Q + \text{encrypted overhead} \).

b) For an intermediate node i on a routing path, the time required to process \( Q' \), denoted by \( T_{d} (i) \), is

\[
T_{d} (i) = \frac{\text{size} (Q')}{\text{Rec_speed(i)} - \text{drop_rate}_{m}(i)}
\]

where \( \text{Rec_speed(i)} \) is node i’s receiving speed, and \( \text{drop_rate}_{m}(i) \) is node i’s arriving packets’ drop rate. Since node i does not decrypt and encrypt \( Q' \), the items of concern are size \( (Q') \), \( \text{Rec_speed(i)} \) and \( \text{drop_rate}_{m}(i) \). Let
\[ L(i) = \text{Rec} \_\text{speed}(i) - \text{drop} \_\text{rate} \_\text{in}(i) \]  

which is the actual receiving speed of node \( i \). Let \( T_{d,\text{in}} \) be total processing time of all intermediate nodes.

\[
T_{d,\text{in}} = \sum_{j=1}^{n} T_d(j)
\]

where \( n \) is the number of intermediate nodes along the underlying routing path.

c) For a destination node \( \text{Des} \), the time required to process \( Q' \), denoted by \( T_{d,\text{d}}(\text{Des}) \), is

\[
T_{d,\text{d}}(\text{Des}) = \frac{\text{size}(Q)}{L(\text{Des})} + \frac{\text{size}(Q')}{\text{decryption speed}(\text{Des})} + \frac{\text{size}(Q)}{\text{processing speed}(\text{Des})}
\]

Let \( T_{d,\text{processing}} \) be the total time required to process \( Q' \).

\[
T_{d,\text{processing}} = T_{d,\text{d}}(\text{src}) + T_{d,\text{in}} + T_{d,\text{d}}(\text{Des})
\]

2) Transmission delay:

Transmission delay is the time period from when the first bit of \( Q' \) is sent out to the time point when the last bit of \( Q' \) is transmitted. The items include are the size of \( Q' \) and actual delivery speed (instead of data rate). An intermediate node on receiving \( Q' \) transmits this packet to next hop without decrypting and encrypting \( Q' \).

A. Let \( T_{d,\text{r}} \) be the transmission delay of the data packet \( Q' \).

\[
T_{d,\text{r}} = \sum_{i=0}^{n} \frac{\text{size}(Q')}{M(i)}
\]

where \( i=0 \) represents the source node \( \text{Src} \), and \( M(i) \), called node \( i \)'s actual delivery speed, is defined as

\[ M(i) = \text{data} \_\text{rate}(i) - \text{drop} \_\text{rate} \_\text{out}(i) \]

where \( \text{drop} \_\text{rate} \_\text{out}(i) \) is node \( i \)'s departing packets' drop rate.

B. Let \( T_{a,\text{r}} \) be the transmission delay of the corresponding ACK.

\[
T_{a,\text{r}} = \sum_{i=0}^{n} \frac{\text{size}(\text{ACK})}{M(i)}
\]

where \( i=0 \) represents ACK's source node \( \text{Des} \).

3) Propagation delay:

Propagation delay is the time period from the time point when the first bit of \( Q' \) is sent out to the time point when the bit is received. The items included are the initial bandwidth and occupied bandwidth. When a packet is transmitted, the propagation delay

\[
T_e = \sum_{(i,j)} \left( \frac{1}{\text{bandwidth}(i) - \text{bandwidth}_{\text{occupied}}(i) - \text{drop} \_\text{rate} \_\text{out}(i)} \right)
\]

where \( \text{bandwidth} \_\text{initial}(i,j+1) \) and \( \text{bandwidth} \_\text{occupied}(i,j+1) \) respectively represent initial bandwidth and occupied bandwidth of the path segment between nodes \( i \) and \( i+1 \), and \( M(i) = \text{bandwidth}(i) - \text{bandwidth} \_\text{occupied}(i) - \text{drop} \_\text{rate} \_\text{out}(i) \).

For an ACK packet, we assume the propagation delay

\[
T_{a,\text{processing}} = T_{a,\text{r}}(\text{Des}) + T_{a,\text{in}} + T_{a,\text{d}}(\text{src})
\]

Let \( T_{d,\text{processing}} \) be total cost for processing an ACK.

\[
T_{a,\text{processing}} = \sum_{i=0}^{n} \frac{\text{size}(\text{ACK})}{M(i)}
\]
and arrival rates, \( \lambda \) and \( \mu \) respectively node \( i \)’s actual arrival and departure rates, \( L(i) \) and \( M(i) \) are respectively node \( i \)’s actual arrival and departure rates,

\[
L'(i) = \frac{L(i)}{\text{packet size}} \quad (18)
\]
\[
M'(i) = \frac{M(i)}{\text{packet size}} \quad (19)
\]

For an ACK packet, we also assume the queuing delay \( T_{\text{a-que}} = T_{\text{d-que}} \) to simplify the scope of the following analyses.

### 3.2 Total cost

Let TC be the total cost of packet delivery

\[
TC = T_{\text{processing}} + (T_{\text{d,r}} + T_{\text{a,r}}) + (T_{\text{d,q}} + T_{\text{a,q}}) + (T_{\text{d,qw}} + T_{\text{a,qw}}) \quad (20)
\]

Since the source nodes and destination nodes of two paths, e.g., paths \( q \) and \( r \), belonging to the same association are themselves the same, and the two paths deliver the same packet \( Q'\) and ACK, the cost difference \( CD_p \) between the two paths \( q \) and \( r \) only results from involving different numbers of intermediate nodes and different intermediate nodes. Hence, according to equations (6), (11), (13), (15), (16) and (17),

\[
CD_p = TC_q - TC_r = (T_{\text{d,in}}^q - T_{\text{d,in}}^r + T_{\text{a,in}}^q - T_{\text{a,in}}^r) + (T_{\text{d,pr}}^q - T_{\text{d,pr}}^r + T_{\text{a,pr}}^q - T_{\text{a,pr}}^r) + 2(T_{\text{d,qw}} - T_{\text{d,qw}}^r) + 2(T_{\text{d,qw}} - T_{\text{d,qw}}) \quad (21)
\]

From equations (2), (3), (4), (8) and (9), we can see that the expression \( (T_{\text{d,in}}^q - T_{\text{d,in}}^r + T_{\text{a,in}}^q - T_{\text{a,in}}^r) \) is a function of \( L(i) \), \( i=1,2,\ldots,n_q+n_r \), where \( n_q \) and \( n_r \) are respectively numbers of path \( q \)’s and path \( r \)’s immediate nodes. Similarly, based on equations (13) and (15), the expression \( (T_{\text{d,pr}}^q - T_{\text{d,pr}}^r + T_{\text{a,pr}}^q - T_{\text{a,pr}}^r) \) is a function of \( M(i) \), \( i=0,1,2,\ldots,n_q+n_r \). Based on equations (16) and (17), the remaining terms are respectively functions of \( M(i) \), and \( L'(i) \) and \( M'(i) \). Let \( CD_q = |CD_p| \), then

\[
CD_q = \sum_{i=1}^{n_q} \frac{\text{size}(Q')}{L_{i,q}} - \sum_{i=1}^{n_r} \frac{\text{size}(Q')}{} + \sum_{i=1}^{n_q} \frac{\text{size}(\text{ACK})}{L_{i,q}} - \sum_{i=1}^{n_r} \frac{\text{size}(\text{ACK})}{L_{i,r}} + \sum_{i=1}^{n_q} \frac{\text{size}(Q')}{M_{i,q}} - \sum_{i=1}^{n_r} \frac{\text{size}(Q')}{M_{i,r}} + \sum_{i=1}^{n_q} \frac{\text{size}(\text{ACK})}{M_{i,q}} - \sum_{i=1}^{n_r} \frac{\text{size}(\text{ACK})}{M_{i,r}} + 2L_{j,q} - 2L_{j,r} + 2M_{j,q} - 2M_{j,r} \quad (22)
\]

where \( L_{j,q} (L_{j,r}) \) is actual receiving speed of node \( j \) on the path \( q \) (actual receiving speed of node \( k \) on path \( r \)), \( M_{j,q} (M_{j,r}) \) is actual delivery speed of node \( j \) on path \( q \) (actual delivery speed of node \( k \) on path \( r \)), \( L'_{j,q} \) and \( M'_{j,q} \) (\( L'_{j,r} \) and \( M'_{j,r} \)) are respectively actual packet arrival rate and departure rate of node \( j \) on path \( q \) (of node \( k \) on path \( r \)), and \( n_q \) and \( n_r \) are respectively numbers of nodes on path \( q \) and \( r \). Note that all summation indexes of equation (22) begin at 1 instead of at 0 due to excluding the source node and destination node.

Since an ACK is a packet of fixed length, given an encrypted packet \( Q' \) and an association that has two paths, \( q \) and \( r \) of lengths \( n_q \) and \( n_r \), respectively, from equations (3), (14), (18), and (19), we can see that only \( \text{Rec_speed}(j,q) \), \( \text{drop_rate}_{\text{in}}(j,q) \), \( \text{data_rate}(j,q) \), \( \text{drop_rate}_{\text{out}}(j,q) \) and \( \text{Rec_speed}(k,r) \), \( \text{drop_rate}_{\text{in}}(k,r) \), \( \text{data_rate}(k,r) \), \( \text{drop_rate}_{\text{out}}(k,r) \) are unknown, \( j=1,2,\ldots,N_q \), \( k=1,2,\ldots,N_r \). On the other hand, if we can access the \( n_q \) and \( n_r \) intermediate
nodes’ network management information through a network management protocol, e.g., Simple Network Management Protocol (SNMP), then we can retrieve the quadruples (Rec_speed(), drop_rate_in(), data_rate(), drop_rate_out()) from all immediate nodes. So, we further assume that all immediate nodes’ management information bases (MIBs) are available, and can be accessed. However, accessing network management information takes time. It is hard to retrieve the information of concern for each path in a real time manner right before choosing a primary path. However, before current accurate information is gathered, we cannot take a right decision and then choose the right path. On the other hand, if we have to access the information before choosing the best path, packet delivery of Q'/Q will be delayed. To solve this problem, we predict the quadruple values for each node by using the exponential average algorithm, $\tau_{m+1} = \alpha \tau_n + (1-\alpha) L_n$, where $\tau_{n+1}$ and $\tau_n$ are respectively the $(n+1)^{th}$ and $n^{th}$ predicted values of one of the quadruple elements, and $T_n$ is the $n^{th}$ real value of the feature retrieved from the corresponding MIB. Here,

$$L_{n+1} = \alpha L_n + (1-\alpha) L_n$$

$$M_{n+1} = \alpha M_n + (1-\alpha) M_n$$

$$\text{drop}_a_{in}^{n+1}(j,q) = \alpha \text{drop}_a_{in}(j,q) \cdot \text{drop}_a_{in}^{n+1}(j,q) + (1-\alpha \text{drop}_a_{in}(j,q)) \cdot \text{drop}_a_{in}^{n+1}(j,q)$$

$$\text{drop}_a_{out}^{n+1}(j,q) = \alpha \text{drop}_a_{out}(j,q) \cdot \text{drop}_a_{out}^{n+1}(j,q) + (1-\alpha \text{drop}_a_{out}(j,q)) \cdot \text{drop}_a_{out}^{n+1}(j,q)$$

where the $L_{n+1}$, $M_{n+1}$, drop$_a_{in}^{n+1}(j,q)$ and drop$_a_{out}^{n+1}(j,q)$ are respectively the $(n+1)^{th}$ (the $n^{th}$) predicted receiving speed, input drop rate, delivery speed and output drop rate of node $j$ on path $q$, $\alpha_{L_{n+1}}$, $\alpha_{M_{n+1}}$, $\alpha_{\text{drop}_a_{in}(j,q)}$, and $\alpha_{\text{drop}_a_{out}(j,q)}$ are respectively weights of node $j$’s receiving speed, input drop rate, delivery speed and output drop rate, and $L_{n+1}$, $M_{n+1}$, drop$_a_{in}^{n+1}(j,q)$ and drop$_a_{out}^{n+1}(j,q)$ are respectively the $n^{th}$ actual receiving speed, actual drop rate, actual delivery speed, actual input drop rate and actual output drop rate. Since SCTP’s path change and switch do not occur frequently, we often have enough time to access the actual values of the four terms from each intermediate node’s MIB.

In an MIB, the items ipInReceives(t) (OID= {1.3.6.1.2.1.4.1.3}), ipInDiscards(t) (OID= {1.3.6.1.2.1.4.1.4}), ipOutRequests(t) (OID= {1.3.6.1.2.1.4.1.10}) and ipOutDiscards(t) (OID= {1.3.6.1.2.1.4.11}) are respectively defined as accumulated numbers of packets that the underlying router has so far received, dropped on the input side, sent and dropped on the output side since the router started up. By retrieving the four items from intermediate node $j$ on path $q$, $L_{n+1}$, $M_{n+1}$, drop$_a_{in}^{n+1}(j,q)$ and drop$_a_{out}^{n+1}(j,q)$, at time point $t_{n+1}$ can be derived, in which the $L_{n+1}$, $M_{n+1}$, drop$_a_{in}^{n+1}(j,q)$ and drop$_a_{out}^{n+1}(j,q)$ respectively in equations (23) ~ (26) can be obtained by accessing the MIB twice at $t_{n+1}$ and $t_n$ right after the previous, i.e., the $n^{th}$ switchover at time $t_n$ by the following equations,

$$\text{ipInReceives}(t_{n+1}) = \text{ipInReceives}(t_{n+1}) - \text{ipInDiscards}(t_{n+1})$$

$$\text{ipInDiscards}(t_{n+1})$$

$$\text{ipOutReceives}(t_{n+1}) - \text{ipOutDiscards}(t_{n+1})$$

$$\text{ipOutDiscards}(t_{n+1})$$

$$\text{ipOutReceives}(t_{n+1}) - \text{ipOutDiscards}(t_{n+1})$$

$$\text{ipOutDiscards}(t_{n+1})$$

in which $t_{n+1} > t_n$. The time point right after the SCTP started up, i.e., the time when the initial primary path has just been selected.

### IV. EXPERIMENTS AND DISCUSSIONS

#### 4.1 Simulation environments

Our simulations were carried out by running a revision of Delaware University’s SCTP module for NS-2. The simulation topology is shown in Figure 1. The two end nodes, sender and receiver, both have 4 IP addresses. Routers 1-1 and 1-2, 2-1 and 2-2, 3-1 and 3-2, 4-1 and 4-2 are routers between the two end nodes. Routers 1-1 is connected to router i-2 and i=1,2,3,4. The bandwidth of path 1 is 2Mbps, and those of paths 2, 3 and 4 are 1.5Mbps, 1.8Mbps and 1Mbps, respectively. The SCTP parameters are all default values except those mentioned above. The sender continuously sends 2 Mbps FTP data to the receiver in a 30-second time period. Switchover occurs at the 10th sec. In the following experiment, we evaluated the PSASP’s end-to-end delays, jitters, throughputs and packet drop rates.

#### 4.2 Simulation results

In the first experiment, three state-of-the-art systems, including the standard SCTP [6], Optimized SCTP [5], and RTT Based SCTP [4], are tested and compared with the PSASP. The default primary path of the standard SCTP is set to path 2.

The experimental results of the four schemes on the end-to-end delays as illustrated in Figure 2 are initially almost the
same. But, after the first switchover, the PSASP had less delay than others. Right after the switchover, the end-to-end delays of the four systems between the 11th and 12th seconds do not increase sharply because they all have enough bandwidth to transmit packets. When time passed and more packets and overheads were sent and involved, respectively, the delays increased quickly. But the PSASP had less delay because it selected the best path. The congestion collapse occurs at the 20th second, causing the reduction of congestion window size. Slow start is then initiated immediately. But Optimized SCTP reduced its congestion window size slowly. That is why after the congestion, its end-to-end delays are still long.

The experimental results of jitters as illustrated in Figure 3 show that the PSASP had smaller jitters than others had. At the point when the primary path begins its transmission, the jitters vibrated because the two sides of the path need to exchange information, resulting in more transmission overheads. However, the transmission and jitters were soon stable. When switchover occurs, the jitters vibrated again, and the other three schemes are larger than they were. The PSASP had a similar phenomenon, but the vibration is smoother and smaller since the PSASP always chooses the best path as the new primary path. Longer transmission delays often result in larger jitters. The congestion collapse occurs at the 20th second, and slow start is initiated. After that, the vibration mitigates since network traffic is lower, and delay is then shorter. But, the Optimized SCTP’s vibration is still huge because its congestion widow decreases slowly when traffic is congested.

The experimental results of packet loss rates are illustrated in Table II. The PSASP exhibits the best also due to choosing the best alternate path, i.e., path 3, which provides a higher transmission quality and stable environment than the default path, i.e., path 2, does.

<table>
<thead>
<tr>
<th>Protocols</th>
<th>sent</th>
<th>received</th>
<th>lost</th>
<th>loss rate(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCTP</td>
<td>4050</td>
<td>4046</td>
<td>4</td>
<td>0.0987</td>
</tr>
<tr>
<td>PSASP</td>
<td>4557</td>
<td>4555</td>
<td>2</td>
<td>0.0438</td>
</tr>
<tr>
<td>Optimized SCTP</td>
<td>4141</td>
<td>4132</td>
<td>9</td>
<td>0.2173</td>
</tr>
<tr>
<td>RTT Based SCTP</td>
<td>3986</td>
<td>3984</td>
<td>2</td>
<td>0.0501</td>
</tr>
</tbody>
</table>

V. CONCLUSION AND FUTURE RESEARCH

In this study, we develop a new path selection and switchover scheme, the PSASP, for the SCTP, which considers the key influential path performance factor, round-trip delay, before selecting a primary path for the SCTP so as to provide the SCTP network transmission with wider bandwidth and a more reliable environment. The round-trip delay is the time required to successfully deliver a packet and receive the corresponding ACK. We further decompose the round-trip delay into processing, transmission, propagation, and queueing delays, and analyze the influential factors of the four delays.

Experimental results show that the PSASP has several significant advantages in terms of lower delays, jitters and drop rates, and higher throughputs over the other three tested schemes.

In the future, we would like to derive the PSASP’s mathematical model of reliability which is a formal model, and study how the considered parameters affect the arrival and service rates of a path segment. So, a user can understand the reliability of the PSASP before using it. Those constitute our future research.

REFERENCES