

# Fast Handover with SCTP Parameter tuning in Wireless LAN

M2011MM082 YOKOGAWA Takahiro  
Supervisor : OKUMURA Yasuyuki

## 1 Introduction

Stream Control Transmission Protocol (SCTP)[6] is a reliable, general-purpose transport layer protocol for IP networks. SCTP provides Multihoming support in which one or both endpoints of a connection can consist of more than one IP address. But SCTP is not designed with Wireless LAN (WLAN) Network. In the case of using SCTP in WLAN, it takes over 170 seconds[2] to make a handover (switching path). This is too long for most applications. To reduce SCTP handover time change two parameters, Path Max Retrans (PMR) and Maximum Retransmission Timeout (RTO.MAX). The purpose of this study is fast handover in wireless LAN. The following approach was employed in SCTP parameter (PMR and RTO.MAX) tuning. However, PMR and RTO.MAX values changing too small is cause of wrong handover. And SCTP parameter tuning was only simulated handover time in NS2.

In this paper, we confirm effectiveness of SCTP parameter tuning with NS2, and emulate with Goto's IP Network Emulator 3 (GINE). Moreover we verify about using CMT-SCTP (Concurrent Multipath Transfer [4]. CMT is the simultaneous transfer of data from a source host to a destination host via two or more end-to-end paths. Evaluation items are handover time and average throughput. We compare default SCTP, CMT-SCTP and SCTP with parameter tuning. The expected results are high-speed and high-quality data sending or receiving.

## 2 Features of SCTP

This section explained about Stream Control Transmission Protocol (SCTP).

### 2.1 SCTP packet structure and connection

TCP and SCTP packet simple structure is Fig1. SCTP packet is composed by common header and any chunk (instead of window).

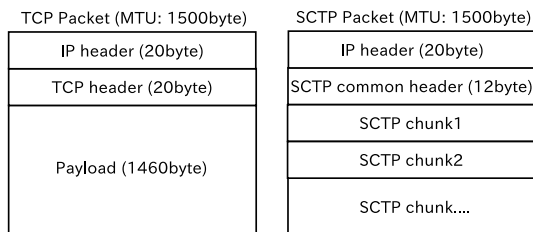


Figure 1 TCP and SCTP packet structure

TCP header length is always 20byte, but SCTP header length is more than 28 byte (12 byte common header length and 16 byte data chunk header length). Table1 shows TCP and SCTP header length and MAX payload.

Table 1 TCP and SCTP packet header length

	TCP	SCTP
header length(Byte)	40	48
MAX payload(Byte)	1460	1452

Additionally, SCTP provides Multihoming support. A multihomed host is one that has more than one network interface and therefore more than one IP address for which it can be addressed. However, default SCTP uses only single path for data sending (called primary path), another paths utilized for backup path (called secondary path).

### 2.2 CMT-SCTP

SCTP cannot use multiple path for data sending concurrently. However CMT-SCTP (Concurrent Multipath Transfer)[4] extension of SCTP can use multipath transfer concurrently. It is supported by only FreeBSD kernel. CMT-SCTP has five different retransmission policies[3][4] that use heuristics for faster loss recovery.

1. RTX-SAME : Once a new data chunk is scheduled and sent to a destination, all retransmissions of the chunk thereafter are sent to the same destination
2. RTX-ASAP : A retransmission of a data chunk is sent to any destination for which the sender has cwnd space available at the time the retransmission needs to be sent. If the sender has available cwnd space for multiple destinations, one is chosen randomly.
3. RTX-CWND : A retransmission of a data chunk is sent to the destination for which the sender has the largest cwnd. A tie is broken by random selection.
4. RTX-SSTHRSH : A retransmission of a data chunk is sent to the destination for which the sender has the largest ssthresh. A tie is broken by random selection.
5. RTX-LOSSRATE : A retransmission of a data chunk is sent to the destination with the lowest loss rate path. If multiple destinations have the same loss rate, then a destination is selected randomly from among them. This policy is an ideal case since the loss rate is known, and not measured in the simulations.

NS2 can use all policies, but FreeBSD can use only RTX-SSTHRESH. In this paper, we use RTX-SSTHRESH for emulation and simulation.

### 2.3 SCTP parameter values

Some SCTP parameters are concern with handover. We show these parameter below (Table2). Table2 is parameters and value in NS2.

Table 2 SCTP parameters (NS2)

Parameter	Param in NS2	default value
PMR	pathMaxRetrans_	5
H.B.Interval	heartbeatInterval_	30000 (30sec)
RTO.MAX	maxRto_	60000 (60sec)
RTO.MIN	minRto_	1000 (1sec)
RTO.Initial	initialRto_	3000 (3sec)
MAX.InitRetrans	maxInitRetransmits_	8
AMR	associationMaxRetrans_	10

SCTP use error count for path management. The error count is incremented when a data timeout occurs on a destination any time. If the error count exceeds PMR, SCTP endpoint considers the address is inactive (Fig2). If the error count exceeds AMR (Association Max retrans), SCTP endpoint considers to terminate association (Fig2).

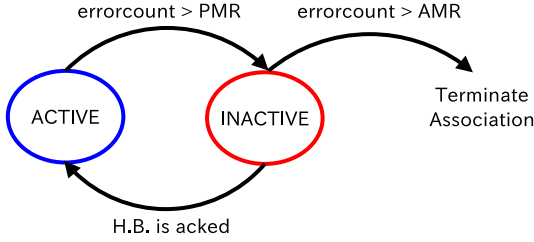


Figure 2 Error count, PMR and AMR

We show the using RTO algorithm in Section2.4. These parameters showed RFC4960[6], but there are not optimum value. We change two parameters value, PMR and RTO.MAX.

#### 2.4 SCTP Backoff algorithm

We show the algorithm to calculate RTO. RTO is very important factor fore handover.

$$RTO = RTO.Initial \quad (1)$$

RTO.Initial (see (1)) is defined as SCTP parameter (Table2), default value is 3[sec]. It is used before Sender get RTT (Round-Trip Time). RTT is used fore SRTT (Smoothed Round-Trip Time) and RTTVAR (Round-Trip Time Variation). If SCTP Sender received RTT, the RTT define as RTT.1st. We show how to calculate SRTT and RTTVAR (show (2)(3)). SRTT, RTT.1st and RTTVAR are initialized as:

$$SRTT = RTT.1st \quad (2)$$

$$RTTVAR = \frac{RTT.1st}{2} \quad (3)$$

RTO is updated to below (4).

$$RTO = SRTT + 4 \times RTTVAR \quad (4)$$

SCTP gets a new measurement of RTT: RTT.new, SRTT and RTTVAR will updated as below (5)(6).

$$RTTVAR.new = (1 - \beta) \times RTTVAR.old + \beta \times |SRTT.old - RTT.new| \quad (5)$$

$$SRTT.new = (1 - \alpha) \times SRTT.old + \alpha \times RTT.new \quad (6)$$

After calculate SRTT and RTTVAR, RTO will updated as below (7).

$$RTO = SRTT.new + 4 \times RTTVAR.new \quad (7)$$

If RTO value is less than RTO.MIN, it set to RTO.MIN value. If RTO value is greater than RTO.MAX, it set to RTO.MAX value. Every time a transmission timeout occurs for an address, the RTO for this address will be doubled below (8).

$$RTO = RTO.old \times 2 \quad (8)$$

If RTO value(8) is bigger than RTO.MAX, it set to RTO.MAX value.

### 3 Network Model

In this section, we explain about the virtual network model (show Fig3). The network model is use for simulation and emulation. Given that the network model use for SCTP. The implementation is configured with no overlap between the two paths.

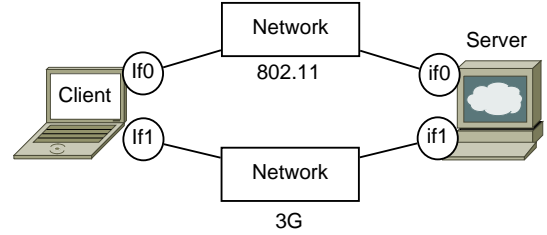


Figure 3 The Network model configuration

We can posit two hypotheses as follows.

1. More than two different wireless standard access points are setting, and the mobile terminal has multiple interfaces for wireless.
2. Mobile terminal can use two wireless standard, Wi-Fi(802.11g) and LTE.

SCTP can use multiple wireless standard as only one association.

#### 3.1 Network delay and throughput

We use measured value of network delay and throughput in simulation and emulation. Mobile terminal to measure is i Phone 4S, and use ping command to measure round-trip delay. Table3 is the results of wireless zone.

Table 3 Network status (wireless)

Wireless LAN Standard	3G	Wi-Fi(802.11g)	LTE
Upload Throughput[Mbps]	0.62	20.79	5.40
Download Throughput[Mbps]	1.03	21.58	10.79
Round-trip Delay[ms]	258.40	63.30	258.40
One-way Delay[ms]	129.20	31.65	129.20

We measure oneway delay and throughput of wired zone too Table4.

Table 4 Network status (wired)

Measurement period	Nagoya to Tokyo
Upload Throughput[Mbps]	100.00
Download Throughput[Mbps]	100.00
Round-trip Delay[ms]	12.26
One-way Delay[ms]	6.13

We use these measured value for simulation and emulation.

### 3.2 Simulation's network model

We show the model of simulation in Fig4. Client means mobile node, Server means the receiver for SCTP client.

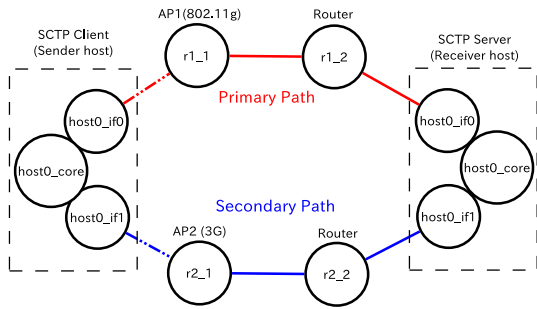


Figure 4 Model of Simulation

If we use SCTP multihomed node in NS2, it needs node for core and node for interfaces. Client in Fig4, host0\_core, host0\_if0 and host0\_if1 show one host. It is same at Server. Between Mobile node (Client) and Access Point (AP) are wireless connection. IF0 and AP1, IF1 and AP2 use different wireless standard. Primary path uses Wi-Fi (802.11g), Secondary path uses LTE. Each path sets values based Table3 and Table4.

### 3.3 Emulation's network model

In this section, we explain about the network model of emulation. To set virtual network, use Goto's IP Network Emulator 3[5](GINE). GINE emulates IPv4/v6 network which component shows router and link. We show the virtual network with GINE for Linux kernel and FreeBSD kernel.

We show the virtual network for FreeBSD kernel below (see Fig5).

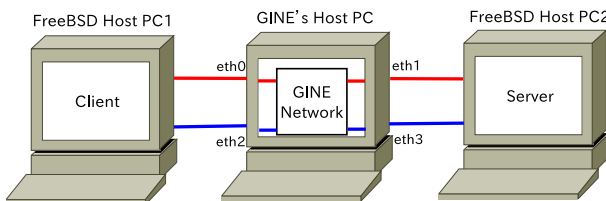


Figure 5 Model of Emulation

Two FreeBSD hosts connect with GINE host (Ubuntu host). We use 2NIC PC for FreeBSD hosts, 4NIC PC for GINE host. Setting of GINE host shows Fig6.

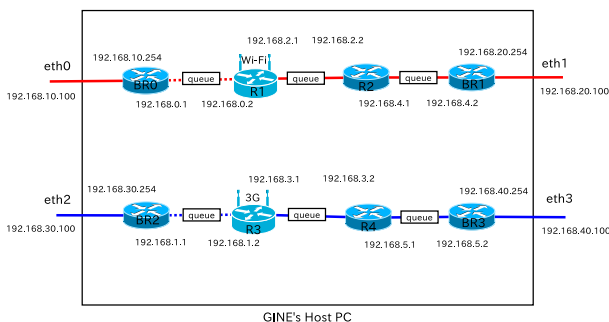


Figure 6 GINE network configuration

BR0, BR1, BR2, BR3 are virtual nodes for connect with bridges. R1 is Wi-Fi wireless router, R3 is LTE wireless router. GINE host and BSD hosts set addresses

by ifconfig command. The network delay and packet loss set values based Table3 and Table4.

## 4 How to Simulate and Emulate

In this section, we show simulation method and emulation method. We can posit two hypotheses as follows.

1. Path is not switched (stable network).
2. Path is switched (network failure).

If the path switched to secondary path, we will measure the time of handover.

### 4.1 Simulation with NS2

We simulate SCTP network, the network model show in Section3.2. We measure the throughput using TCP, SCTP, and CMT-SCTP. Client (Section3.2) begins to send data to Server after 0.5 seconds. (Sending data method is ftp.)

The NS2 simulation scenario is the below.

1. After NS2 scenario is start, Sender and Receiver connect by two paths each other.
2. IF0 (primary path) is down after 100.0 seconds.
3. NS2 scenario finish after 300.0 seconds.
4. We measure switchover time and throughput.

### 4.2 Emulation with GINE

We explain about emulation method and programs to send or receive SCTP packets. Emulation network model show Section3.2. We measure the throughput using TCP, SCTP, and CMT-SCTP. We explain about the programs using FreeBSD socket. CMT will considered for implementation as Internet draft[1], but it can use in only FreeBSD. Therefore we use FreeBSD to emulate CMT-SCTP. We change the Linux socket programs to send or receive and calculate throughput for FreeBSD socket programs. Emulation scenario is the below.

1. Ubuntu host start GINE program, and configure GINE emulation network.
2. BSDhost2 (server) start the server program.
3. BSDhost1 (client) start the client program.
4. We remove the address 192.168.10.1 at BSDhost1.
5. We measure switchover time and throughput.

## 5 Results of simulation and emulation

We show the results of NS2 simulation and GINE emulation.

### 5.1 Evaluation of SCTP on stable network

We show the throughput of TCP, SCTP and CMT-SCTP on simulation(Table5) network. We suppose the SCTP network has no handover.

Table 5 Throughput[Mbps] (stable network)

Protocol	TCP	SCTP	CMT-SCTP
NS2 Throughput[Mbps]	19.62	19.27	20.95
GINE Throughput[Mbps]	19.94	19.91	20.62

Table 5 shows mean values of 10 times measurement for handover times 95% on GINE emulation. Confidence intervals are omitted because they are very narrow relative (less than 0.5%) to those mean.

## 5.2 Evaluation of SCTP on the switchover network

We show the results of simulation and emulation on the network switchover network. In this section we explain about SCTP parameter tuning and using CMT-SCTP. Table6 illustrates the handover times for PMR values 5 to 1 when the RTO.MAX is set to 60, 30, 10, 5, and 2 second respectively. We measure handover time and throughput ten times, and calculate 95% confidence interval on GINE emulation.

Table 6 Handover times[sec] (switchover network)

		RTO.MAX				
		60	30	10	5	2
PMR	5	66.43	56.37	38.43	25.43	9.43
		64.99	62.96	36.97	23.97	12.96
	4	33.69	33.69	27.69	19.69	7.69
		32.98	33.31	26.97	18.97	10.70
	3	16.95	16.95	16.95	13.95	5.95
		16.97	16.97	16.96	13.97	8.97
	2	8.21	8.21	8.21	8.21	4.21
		8.97	8.61	9.91	8.97	5.94
	1	3.48	3.48	3.48	3.48	2.48
		4.98	4.90	5.33	4.98	4.21

upper: NS2, lower: GINE

When RTO.MAX value is 1, SCTP don't switchover secondary path. Table6 shows the best PMR value is 1 and RTO.MAX value is 1 in WLAN environment. Table7 illustrates the throughput for PMR values 5 to 1 when the RTO.MAX is set to 60, 30, 10, 5, and 2 second respectively.

Table 7 Throughput[Mbps] (switchover network)

		RTO.MAX				
		60	30	10	5	2
PMR	5	11.31	11.38	12.23	12.67	13.20
		9.07	9.10	9.86	10.04	10.19
	4	12.38	12.38	12.58	12.84	13.26
		9.96	9.98	10.16	10.40	10.49
	3	12.92	12.92	12.92	13.02	13.32
		10.18	10.45	10.45	10.54	10.73
	2	13.20	13.20	13.20	13.20	13.37
		10.18	10.35	10.64	10.69	12.27
	1	13.38	13.38	13.38	13.38	13.41
		10.58	10.55	10.59	10.60	12.32

upper: NS2, lower: GINE

Table 6 and 7, shows mean values of 10 times measurement for handover times 95% on GINE emulation. From the Table6 and 7, the results indicate the default PMR and RTO.MAX value are not sufficient to guarantee an aggressive switch strategy. By setting default delays of 64.99 seconds were experienced before switchover occurred, and PMR=2 delays of 4.21 seconds were experienced before switchover occurred. The results illustrates that minimum values for both PMR and RTO.MAX should be employed in WLAN.

Table 8 SCTP throughput[Mbps] (switchover network)

Protocol	SCTP	SCTP(tuning)	CMT-SCTP
NS2 [Mbps]	11.31	13.41	13.59
GINE[Mbps]	9.07	12.32	12.46

Table8 shows using CMT-SCTP is a little more effective than parameter tuning. CMT-SCTP uses multi-path for data sending, and has no handover time. However, if the paths have nearly delay or packetloss, CMT-SCTP is more effective in multihome network[7]. Table9 is one of the example, two paths set bandwidth 100Mbps, round-trip delay set 11ms.

Table 9 Throughput[Mbps] (stable network)

Protocol	TCP	SCTP	CMT-SCTP
GINE [Mbps]	93.54	91.87	144.97

In the Emulation (Table9), CMT-SCTP transport more than 100Mbps. Depending on the situation, CMT-SCTP get higher throughput than the others.

## 6 Conclusion

This paper shows that SCTP parameter tuning is effective for shortening handover time. We simulate and emulate SCTP parameter (RTO.MAX and PMR) tuning and CMT-SCTP in WLAN environment. As the result, overall throughput in WLAN environment with frequent impediments is shown to be improved by the tuning in NS2 simulation and GINE emulation. If the mobile terminal supports CMT-SCTP, it get higher throughput than parameter tuning. We show the future works below.

1. SCTP other parameter tuning and path management in WLAN.
2. Emulate WLAN network which has more than two paths.
3. Emulate WLAN network that paths has packet loss.
4. Using CMT-SCTP and parameter other parameter tuning.

## References

- [1] Amer P., Becke M., Dreiholz T., Ekiz N., et al., "Load Sharing for the Stream Control Transmission Protocol (SCTP)," RFC 6458 (Informational), 2012.
- [2] Fallon S., Jacob P., Qiao Y., Murphy L., et al., "SCTP Switchover Performance Issues in WLAN Environments," In *Consumer Communications and Networking Conference, CCNC 2008. 5th IEEE*, pp. 564–568, 2008.
- [3] Iyengar J., Janardhan R., Amer R., Paul D. and Stewart, R., "Retransmission policies for concurrent multipath transfer using SCTP multihoming," In *Networks, 2004. (ICON 2004). Proceedings. 12th IEEE International Conference on*, Vol. 2, pp. 713–719 vol.2, 2004.
- [4] Janardhan R., Paul D. and Stewart R., "Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths," *IEEE/ACM Trans. Netw.*, Vol. 14, No. 5, pp. 951–964, 2006.
- [5] Goto K., "Network Emulator with Virtual Host and Packet Diversion," In *Proc. of Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT), Vol. 3, No. 3*, pp. 13–20, 2012.
- [6] Stewart R., "Stream Control Transmission Protocol," RFC 4960 (Proposed Standard), 2007.
- [7] 清水祐介, "SCTP-CMT による複数経路の実験," 南山大学情報理工学部卒業論文要旨集, 2013.