

# Receiver-driven Cooperation-based Concurrent Multipath Transfer over Heterogeneous Wireless Networks

Yuanlong Cao<sup>1</sup>, Qinghua Liu<sup>1</sup>, Yi Zuo<sup>2</sup> and Minghe Huang<sup>1</sup>

<sup>1</sup> School of Software, Jiangxi Normal University, Nanchang, 330022 - China  
[e-mail: ylcao@jxnu.edu.cn, 004799@jxnu.edu.cn, huangmh0093@sina.com]

<sup>2</sup> Jiangxi innovation funds management center for small and medium-sized enterprises  
Nanchang, 330046 – China  
[e-mail: long-1984@163.com]

\*Corresponding author: Yuanlong Cao

*Received December 21, 2014; revised March 25, 2015; accepted June 19, 2015;  
published July 31, 2015*

---

## Abstract

The advantages of employing SCTP-based Concurrent Multipath Transfer (CMT) have been demonstrated to be very useful for data delivery over multi-homed wireless networks. However, there is still significant ongoing work addressing some remaining limitations and challenges. The most important concern when applying CMT to data delivery is related to handling packet reordering and buffer blocking. Another concern on this topic is that current sender-based CMT solutions seldom consider balancing the overhead and sharing the load between the sender and receiver. This paper proposes a novel Receiver-driven Cooperation-based Concurrent Multipath Transfer solution (CMT-Rev) with the following aims: (i) to balance overhead and share load between the sender and receiver, by moving some functions including congestion and flow control from the sender onto receiver; (ii) to mitigate the data reordering and buffer blocking problems, by using an adaptive receiver-cooperative path aggregation model, (iii) to adaptively transmit packets over multiple paths according to their receiver-inspired sending rate values, by employing a new receiver-aware data distribution scheduler. Simulation results show that CMT-Rev outperforms the existing CMT solutions in terms of data delivery performance.

---

**Keywords:** Multipath transport protocol, SCTP, Concurrent Multiple Transfer, cooperative communication, load balancing

---

This research was supported by the Science and Technology Research Project of Jiangxi Provincial Department of Education(GJJ14246), by the Higher School Teaching Reform Research Subject of Jiangxi Province (JXJG-13-2-1), and by the Doctoral Fund of Jiangxi Normal University (6630).

## 1. Introduction

In recent years, various wireless access technologies, such as WiFi, UMTS, WiMAX, LTE, etc., have undergone extremely rapid development. Promoted by the latest technological advances, more and more wireless devices and handheld mobile devices (i.e., personal digital assistants (PDAs), smart phones, etc.) are equipped with multiple network interfaces [1-2]. Such multi-homed devices have multiple heterogeneous access capability, they can transmit data by using more than one additional path (i.e., a secondary path) as an alternative to a primary path and increase path redundancy, enabled by the Stream Control Transport Protocol (SCTP) [3]. With its attractive features of multi-homing and multi-streaming, the SCTP has been recognized as a desired transport layer protocol to provide seamless, continuous and high quality data delivery service under stringent bandwidth, delay, and loss heterogeneous wireless networks [4-5].

Concurrent multipath transfer (CMT) [6-7] uses SCTP's multi-homing feature to concurrently send SCTP chunks across multiple independent end-to-end (e2e) paths in a SCTP association. Using the CMT's parallel transmissions and bandwidth aggregation features, a wireless device equipped with multiple network interfaces can increase the efficiency of data transmission, maximize the network resource utilization, and improve system robustness. Fig. 1 illustrates a basic CMT usage in a heterogeneous wireless network environment. It shows how the multi-homed wireless devices can simultaneously use two paths (path 1 and path 2) to communicate with the multimedia server. This approach improves the connection reliability and protects against the one-path-related errors and failures, common in wireless transmission. Therefore, CMT has been regarded as the ideal technology for content-rich time-sensitive multimedia distribution in heterogeneous wireless networks [8-9].

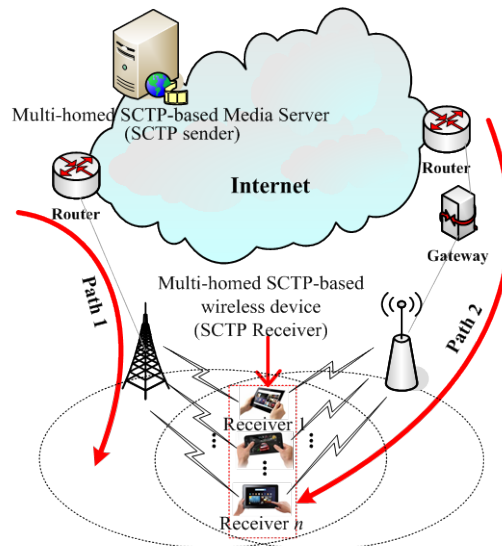


Fig. 1. CMT-based multimedia delivery over a heterogeneous wireless network

Although CMT-based data delivery has gained a variety of attentions and the growing interest in this area has resulted in thousands of peer-reviewed publications, there is still

significant ongoing work addressing many remaining limitations and challenges. The most important concern when applying CMT to data delivery is related to handling packet reordering and buffer blocking. Due to the dynamic and difference of asymmetric path characteristics (e.g., loss rate, e2e delay, bandwidth) [10], the classic CMT's round-robin-based data scheduling mechanism is bound to buffer blocking, with large numbers of out-of-order packet arrivals and severe packet reordering in the constrained receive buffers [11]. Worse for the time-sensitive multimedia applications, the buffered multimedia data may not be handed over to application layer before their playout time and become useless [12]. Although many CMT efforts have been devoted to addressing this issue, they do not take into account the impacts from the receiver. Their sender-based AIMD (additive-increase/multiplicative-decrease)-like sending rate controller, inherited from the standard SCTP, may lead to abrupt and frequent transmission rate fluctuation by overly cutting the congestion window (cwnd) down.

Another important limitation of the existing works on CMT-based data delivery is that they seldom consider balancing the overhead and sharing the load between the sender and receiver. They depend solely upon the sender to run congestion control, sending rate adjustment, path quality evaluation. However in some cases, alike Fig. 1, the sender (media server) may become bottleneck in terms of calculating and scheduling once having a large amount of receivers (*Receiver 1, ..., Receiver n*) communicated concurrently [13]. Some efforts [14-15] shift some functions, such as data sending rate control, path selection and switching decision, from the sender onto receiver. On one hand, running some operations at receiver is regarded as a promising solution for load balancing between the sender and receiver. On the other hand, the receiver runs some functions like sending rate control can make it do not feedback to the sender the network parameters obtained in its forward paths but rather immediately use the first-hand information to determine its desired sending rate. Unfortunately, these solutions do not take into consideration any of the benefits brought by CMT.

This paper proposes a novel Receiver-driven Cooperation-based Concurrent Multipath Transfer solution (CMT-Rev) for efficient multipath parallel data distribution over heterogeneous wireless networks. The goals of CMT-Rev are (i) to balance overhead between the sender and receiver, (ii) to make the sender be aware of receiver's desired sending rate, (iii) to provide the sender with a proper data scheduling scheme, (iv) to alleviate the packet reordering problem and mitigate CMT's buffer blocking, and (v) to improve the CMT throughput performance. The proposed CMT-Rev was thoroughly tested and results showed how it outperforms existing solutions in terms of performance and quality of service.

The rest of the paper is organized as follows. In Section 2 a brief description of related work and the contributions of our CMT-Rev solution are given. Section 3 details the CMT-Rev design. Section 4 evaluates and analyzes the performance of the CMT-Rev. Section 5 concludes the paper and gives our future work.

## 2. Related Work

In the recent years, CMT has gained extensive research interests. Budzisz *et al.* [4] investigated and summarized SCTP and CMT-related articles by developing a four-dimensional taxonomy reflecting the (1) protocol feature examined, (2) application area, (3) network environment, and (4) study approach. They gave a clear perspective on this research area covering both current and future research trends. Wallace *et al.* [16] presented a comprehensive review of the SCTP and further examined three main research areas including (1) handover support and management, (2) CMT-based load sharing, and (3) cross-layer

activities. Dreibholz *et al.* [17] introduced the on-going SCTP standardization progress in the IETF and addressed an overview of future standardization activities and challenges in the areas of SCTP's concurrent multipath transport extension. CMT has been recognized as one of the hot research topics in the context of multi-homed SCTP-based wireless networks.

Iyengar *et al.* [6] were first to design CMT and its related effective algorithms aiming at efficient CMT operations. They also identified three main challenges of CMT including (1) unnecessary fast retransmission, (2) overly conservative cwnd growth at a sender, and (3) increased acknowledgment (ack) traffic. Huang *et al.* [18] developed a fast retransmission strategy dubbed as RG-CMT to deal with packet loss for SCTP-based vehicular networks, supported by the use of relay gateways for CMT. When the SCTP packets are lost due to wireless error or handover, RG-CMT is able to fast retransmit them from the relay gateway to the vehicle timely. Wang *et al.* [19] proposed a wireless CMT SCTP (WCMT-SCTP) to solve the received buffer blocking problem and improve the system throughput in SCTP-based ad hoc networks. However, all above solutions uses the CMT's round-robin scheduler to send packets equally over all the paths, despite their very likely different handling capacities.

Lately, there has been increasing interest in the research on CMT-based multimedia distribution. Xu *et al.* [9] developed a novel evaluation tool-set to analyze and optimize the performance of SCTP CMT-based multimedia content delivery. Huang *et al.* [10] proposed a partially reliable CMT for efficient multimedia delivery by jointly applying the techniques of SCTP's partial reliability extension, prioritized stream transmission and CMT. Xu *et al.* [12] proposed a novel cross-Layer fairness-driven SCTP-based CMT solution (CMT-CL/FD) for parallel video transfer in heterogeneous wireless network environment. CMT-CL/FD improves users' experience of quality for multimedia streaming service while still remaining fair to the competing TCP flows. Baek *et al.* [20] designed a lightweight SCTP-based video partially reliable multicast solution appropriate for mobile wireless network environment. However, the authors ignored the fact that the CMT-based video delivery performance was overly degraded by data reordering due to the path quality differences.

More recently, many researchers have concentrated their efforts to tackle the data reordering issue. Wallace *et al.* [21] designed a renewal congestion window management theory and Markov chain-based analytical framework to model the desired CMT throughput performance. Xu *et al.* [22] developed a generic quality-aware adaptive CMT solution (CMT-QA) for data delivery over heterogeneous wireless networks. CMT-QA distributes SCTP packets over multiple paths according to their own handling capabilities in order to ensure the received data arrives in order. Our previous work CMT-CC [23] proposed a cross-layer cognitive scheduler for CMT necessitating the following aims: (1) alleviate the data reordering problem, (2) improve the CMT performance and quality of service, and (3) fairness to TCP-like flows. Perotto *et al.* [24] extended the CMT's round-robin scheduler based on two types of bandwidth estimations (i.e. Packet Pair and TCP Westwood+), and chooses the paths with lowest transmission time for data transmission. Becke *et al.* [25-26] applied the idea of Resource Pooling to CMT in order to achieve a desired performance over non-CMT transfer while still remaining fair to the competing flows on congested links. However, all above solutions use a "Full CMT" model-which means scheduling SCTP packets over all available paths for data delivery. They depend solely upon the sender to evaluate path quality and lack any consideration for load balancing between the sender and receiver. Our CMT-Rev solution makes fundamental contributions against the state of art in the literature, in the following aspects:

- It provides a newly receiver-driven path quality estimation mechanism to accurately determine each path's sending rate at receiver.
- It designs an adaptive receiver-cooperative path aggregation strategy to assemble a subset of suitable paths for parallel transmission and bandwidth aggregation.
- It introduces an innovative receiver-aware data scheduling algorithm to reduce buffer blocking problems and improve data delivery performance.

### 3. CMT-Rev Detail Design

Fig. 2 illustrates the architecture of CMT-Rev system. CMT-Rev consists of three major blocks, which are *Receiver-driven Sending Rate Controller (SRC-*rev*)*, *Receiver-cooperative Path Aggregation Model (PAM-*rev*)*, and *Receiver-aware Data Distribution Scheduler (DDS-*rev*)*.

- SRC-*rev* runs at CMT-Rev receiver aiming to monitor each path's sending rate constantly, determine an Appropriate Sending Rate (ASR) value, and timely feedback the ASR value to the sender.
- PAM-*rev* serves to select favorable paths to trigger *Full CMT*-which means scheduling packets over all available paths, or *Partial CMT*-which means scheduling packets over partially selected paths for multipath parallel transmission.
- DDS-*rev* is devoting to adaptively adjust sending rate for each path according to their corresponding ASR values. It also intelligently schedules SCTP packets over multiple paths according to their ASR values.

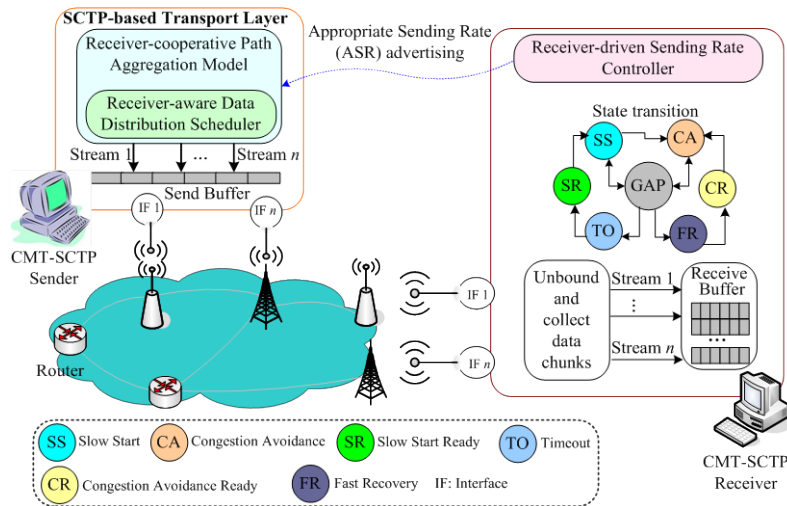


Fig. 2. CMT-Rev architecture

There are seven states running at CMT-Rev's receiver, which are Slow Start, Congestion Avoidance, Slow Start Ready, Timeout, Congestion Avoidance Ready, and Fast Recovery. As they are inherited from the TEAR solution [15], the state transition among the seven states is same as that of TEAR. Like TEAR [15], the CMT-Rev receiver uses the conception of *round* to measure the value of Round-Trip Time (RTT) and Retransmission Timeout (RTO). A *round* begins when a SACK (selective acknowledgment) arrives at the CMT-Rev sender and the sender uses the ASR value carried in the SACK chunk for data delivery. Current round ends and a new round begins once having a new SACK chunk arriving at sender. In CMT-Rev, each

SCTP packet will carry a 4 bits Round ID (RID) to help the CMT-Rev receiver identify the round information. The motivation of using 4 bits for RID is inspired by the designed 4 bits path identifier [27], which is also defined in the SCTP data chunk for identifying the path. Fig. 3 presents the format of extended SCTP data chunk used in CMT-Rev.

Type=0x00	Flags=UBE	Chunk Length
RID	TSN	
Stream Identifier		Stream Sequence Number
Payload Protocol Identifier		
Variable Length User Data		

Fig. 3. Format of extended Data chunk used in CMT-Rev

The CMT-Rev receiver records the timestamp  $t_i$  when it sends a SACK chunk  $S_i$  to the sender. Once  $S_i$  is received, the CMT-Rev sender adjusts its sending rate according to the ASR value advertised in  $S_i$  for data delivery, which means that a new round  $Round_j$  starts. Once the first packet with the RID  $Round_j$  arriving, the CMT-Rev receiver records the timestamp  $t_{i+1}$  and runs the RTT calculation by

$$RTT = \alpha \times RTT' + (1 - \alpha) \times (t_{i+1} - t_i), \quad (1)$$

which  $RTT'$  stands for current round trip time,  $\alpha$  is weighting parameter with a default value 1/8 [3]. The RTO calculation at CMT-Rev receiver is same as that does at the classic CMT sender.

### 3.1 Receiver-driven Sending Rate Controller (SRC-rev)

As previous mentioned, the SCTP, also CMT's AIMD-like sending rate adjustment strategy may occur bursty transmission fluctuation in lossy wireless transmission. As a remedy, our previous SCTP-Rev solution [13] includes a receiver-based sending rate estimator (SRE-rev) running at receiver to provide the sender with a smooth sending rate aiming to avoid bursty transmission fluctuation while maximizing the resource utilization. The CMT-Rev's SRC-rev actually inherits SCTP-Rev's SRE-rev module.

Let's suppose  $x$  possible paths  $(d_1, d_2, \dots, d_x)$  within the SCTP association. Taking path  $d_y$  ( $1 \leq y \leq x$ ) for example, when a packet delivered on  $d_y$  is successfully received by receiver, CMT-Rev enables the SRC-rev to estimate the sending rate for  $d_y$  by [13]

$$rate_{i+1}^{d_y} = \begin{cases} \Phi \times rate_i^{d_y} + (1 - \Phi) \times \frac{1}{N} \times \frac{L_{size}}{RTT}, & \text{if } PLR < thresh; \\ (1 - \Theta) \times rate_i^{d_y} + \Theta \times \frac{1}{N} \times \frac{L_{size}}{RTT}, & \text{otherwise,} \end{cases} \quad (2)$$



where  $N$  denotes the number of received packets by receiver within a round.  $L_{size}$  is the size of the received packets. The weighting factors  $\Phi$  and  $\Theta$  use a default value  $\frac{1}{2}$  for the sake of fairness [13].

In order to make the paper self-contained, we here simply introduce the benefits brought by above rate control behavior. Compared with the AIMD-like rate control behavior, it is found that (i) for network congestion condition, since  $\Delta rate = \frac{1}{N} \times \frac{L_{size}}{RTT} \approx 0$ , we have  $rate_{i+1}^{dy} \approx \frac{1}{2} \times rate_i^{dy}$ . In this case, SRC-*rev* executes the rate control same as SCTP CMT's AIMD-like behavior (by cutting the congestion window in half [28]); (ii) as for non-congestion condition or consecutive congestion condition, supposing  $rate_i^{dy} \approx \frac{1}{N} \times \frac{L_{size}}{RTT}$  once the gap between  $rate_i^{dy}$  and  $\frac{1}{N} \times \frac{L_{size}}{RTT}$  is less than a standard value. Thus, the SRC-*rev* can adaptively tune the rate value up to the maximum or down to the minimum appropriate to the detected condition. Such rate control behavior makes CMT-*Rev* to not only prevent bursty transmission fluctuation but also efficiently utilize the available bandwidth resource.

Before advertising the sending rate value to the sender, the receiver further smoothes it by

$$Rate_e^{dy} = \frac{\bar{Y} \times \Xi_g}{\sum_{s=1}^{\kappa} \gamma_s} \quad (3)$$

where  $\Xi_g$  ( $\Xi_g = [rate_{g,1}, rate_{g,2}, \dots, rate_{g,\kappa}]$ ) denotes the matrix used to store the sending rate values for each path.  $\bar{Y}$  ( $\bar{Y} = [\gamma_1, \dots, \gamma_s, \dots, \gamma_\kappa]$ ) is a weighted coefficient vector, where  $\gamma_s$  ( $s \in [1, \kappa]$ ) can be expressed by

$$\gamma_s = \begin{cases} 1, & \text{if } 1 \leq s \leq \frac{\kappa}{2}; \\ 1 - \frac{s - \kappa/2}{\kappa/2 + 1}, & \text{if } \frac{\kappa}{2} \leq s \leq \kappa. \end{cases} \quad (4)$$

Like [13], CMT-*Rev* launches a timer with 1 RTT in length at receiver for each path as long as a new round starts. Once a timeout event occurs, or the sending rate used for data delivery by the sender is larger than  $Rate_e^{dy}$ , the receiver advertises the estimated  $Rate_e^{dy}$  value to the sender by a SACK chunk. To make it happen, we extend the SCTP SACK and HEARTBEAT ACK, which are presented in Fig. 4 (a) and (b), respectively. The two extensions include three additional parameters, which are Timestamp that aims to order the SACKs received from the asymmetric paths, *pid* (namely path identifier) that is devoting to specifying path between the sender and receiver [27, 29], and the ASR value that serves to offer the sender the sending rate.

Type=3	Chunk Flags	Chunk Length
Cumulative TSN Ack		
Advertised Receiver Window Credit (a_rwnd)		
<b>Timestamp</b>		
<i>Pid</i> #1	Appropriate Sending Rate ( $Rate_e^{d_1}$ )	
<i>Pid</i> #x	Appropriate Sending Rate ( $Rate_e^{d_x}$ )	
Number of Gap Ack Blocks =N		Number of Duplicate TSNs =X
Gap Ack Block #1 Start		Gap Ack Block #1 End
....		
Gap Ack Block #N Start		Gap Ack Block #N End
Duplicate TSN 1		
....		
Duplicate TSN X		

Fig. 4. Format of extended SACK chunk used in CMT-Rev

### 3.2 Receiver-cooperative Path Aggregation Model (PAM-rev)

As mentioned previously, the classic CMT technology mainly adopts a round-robin strategy to equally split packets over all available paths within a SCTP association. Such simple “Full CMT” multipath transmission model and “equal-share” data scheduling way do not consider the fact that asymmetric paths may be with disparate characteristics, they may result in out-of-order data buffered in the constrained receiver buffer and cause serious problems in data delivery.

Taking Fig. 5 (a) and (b) for example, since the packets with TSN (Transmission Sequence Number) 3-4 at path 1 cannot arrive at the receiver at the time  $t_i$  due to path quality difference in terms of delay in Fig. 5 (a) or loss in Fig. 5 (b). The packets with TSN 1-2 and TSN 5-6 have to be buffered in the receiver buffer for reordering and cannot be handed over to application layer. Current mobile devices and handheld mobile devices (i.e., smart phones and PDAs) commonly have very constrained memory and limited free space for the receiver buffer [22]. With severe out-of-order packet arrivals and large amount of data reordering in the overloaded buffer, the classic CMT-based data delivery will undoubtedly suffer the buffer blocking problem.

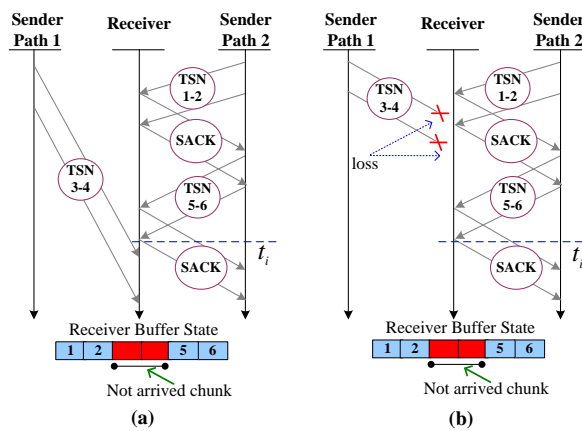


Fig. 5. Out-of-order caused by (a) delay or (b) packet loss



Although many efforts are devoting to CMT's data reordering problem, they still have some limitations on this topic: (i) they actually use a path quality-aware scheduler for data delivery in a "Full CMT" way-which means splitting SCTP packets over all available paths, (ii) they mostly adopt a "sender-depended" path quality evaluation and seldom consider to share the calculation load between the sender and receiver. As a remedy, we in this section design PAM-*rev* module to provide CMT-*Rev* with an adaptive "Full CMT-Partial CMT" interchange model in order to make CMT-*Rev* reduce the blocking problem while achieving load balancing between the sender and receiver.

Using PAM-*rev*, the sender does not calculate transmission efficiency for each path. This feature makes CMT-*Rev* possible reduce the sender's overhead. The CMT-*Rev*'s sender just sorts the paths in a descending order according to their ASR values provided by the receiver. As analyzed above, the paths with huge quality differences is bound to buffer blocking. Meanwhile, the ASR value of each path is mainly determined by path's delay and loss, and can act as a metric to reflect path quality. Therefore, PAM-*rev* is aware of paths' ASR value, and it selects a subset of favorable paths to construct an optimal candidate path list (denoted  $d_{list}$ ) for bandwidth aggregation and load sharing.

Supported by the PAM-*rev* module, when receive buffer blocking is detected, the CMT-*Rev* sender starts a "Partial CMT" model by deactivating the path  $d_j$  that minimizes the following objective function for data delivery,

$$d_j^* = \arg \min_{d_j} Rate_e^{d_j}, \quad (5)$$

subject to

$$\begin{cases} d_j \in d_{list}; \\ 1 \leq j \leq x. \end{cases} \quad (6)$$

If one or more deactivated paths have larger ASR value than that of any one of the paths within the  $d_{list}$ , they will be inactivated by the PAM-*rev* module for data delivery, which means, they will be putted into the  $d_{list}$ . Such "Full CMT-Partial CMT" interchange model helps CMT-*Rev* reduce buffer blocking problem while efficiently aggregating bandwidth resource for concurrent multipath data transfer.

Moreover, PAM-*rev* includes an improved receiver-based sending rate-aware scheduler dubbed as DDS-*rev* to possibly support in-order packet arrival. Once having packets to send, DDS-*rev* will

- (i) compare and sort the paths in a descending order according to their ASR values;
- (ii) use the first path (denoted  $d_{list(0)}$ ) in the  $d_{list}$  as the candidate path (denoted  $d_{send}$ ) to send the packets; or
- (iii) select the next path in the  $d_{list}$  as  $d_{send}$  when  $d_{list(0)}$ 's cwnd is full, and so on.

The pseudo code of DDS-*rev* module-based data distribution algorithm is presented in Algorithm 1.

**Algorithm 1:** receiver-aware data distribution algorithm

Definition:

 $d_{list}$  : a subset of suitable paths selected by PAM-*rev* $d_i$  : the  $i^{th}$  path within the  $d_{list}$  $d_{list(0)}$  : the first path within the  $d_{list}$  $d_{send}$  : a candidate path used for data delivery

When there is a packet to send,

- 1: **for** all paths  $d_i$  within the  $d_{list}$  **do**
- 2:   sort the paths in a descending order according to their ASR values;
- 3:   set  $d_{send} = d_{list(0)}$  ;
- 4: **end for**
- 5: **while** the cwnd of  $d_{list(0)}$  is full **do**
- 6:   set  $d_{send} = d_{list(0)} \rightarrow next$  ;
- 7: **end while**
- 8: schedule the packet over  $d_{send}$  .

## 4. Simulations and Analysis

### 4.1 Simulation topology

The performance evaluation has been carried out on the well-known Network Simulator version 2.35 (NS 2.35) [30]. The simulations considered the heterogeneous wireless network environment shown in Fig. 6. Both SCTP endpoints have three paths (denoted Path 1, Path 2, and Path 3) with different networking parameters. Path 1's bandwidth is set to 11Mbps and 10-20 ms propagation delay, which corresponds to a WiFi/IEEE 802.11b link. Path 2's bandwidth is set to 10 Mbps and 10-20 ms propagation delay, which is representative for a WiMax/IEEE 802.16 link. Path 3 experiences 2 Mbps bandwidth with 10-20 ms propagation delay which is encountered in WiFi/IEEE 802.11 standard. The major configuration of the three paths is presented in Table 1. The receive buffer (rbuf) is set to the default 64KB. The other SCTP default parameters just use the default value in NS 2.35. The total simulation time is 120 seconds.

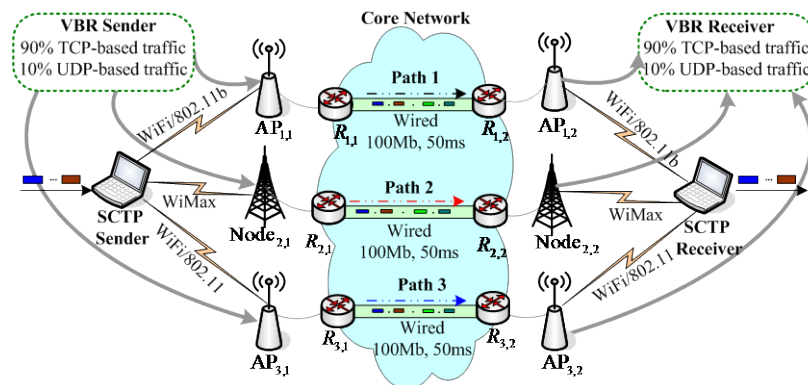


Fig. 6. Simulation topology

To simulate the wireless loss at data-link layer, we attach the uniform loss model and Two-State Markov loss model for each wireless link, to represent distributed loss occurred by contention and infrequent continuous loss caused by signal fading, respectively. Moreover, we inject Internet background traffic generated by a Variable Bit Rate (VBR) generator with Pareto distribution, which sends VBR traffic to its corresponding VBR receiver over the three paths. Like [22], the packets sizes used for the background traffic are selected as follows: 50% are 44 bytes long, 25% are 576 bytes, and 25% have 1500 bytes long. 10% of these packets are carried by UDP protocol and the rest 90% are over TCP protocol. The aggregate background traffic on each path experiences randomly between 0-50% of the access link bandwidth.

**Table 1.** Path configuration used in the simulation

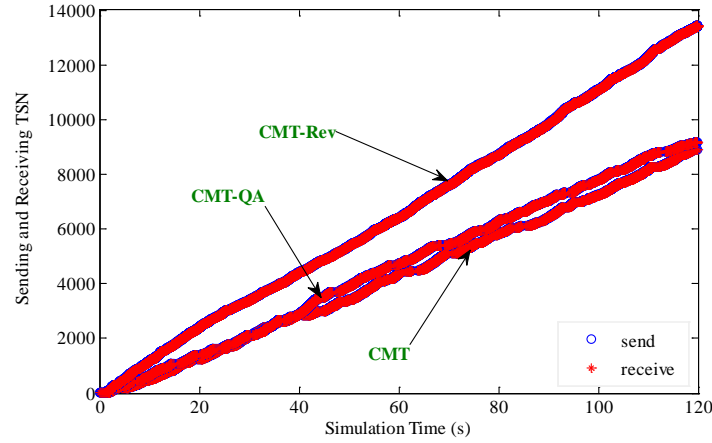
Parameters	Path 1	Path 2	Path 3
Wireless technology	IEEE 802.11b	IEEE 802.16	IEEE 802.11
Access link bandwidth	11Mbps	10Mbps	2Mbps
Access link propagation delay	10-20ms	10-20ms	10-20ms
Access link queue limit	50	50	50
Access link queue type	Droptail	Droptail	Droptail
Uniform loss rate	1-2%	1-2%	5-6%
Markov loss rate <sup>1</sup>	1%	1%	1%
Core network propagation delay	50ms	50ms	50ms

1. Markov loss model is a Markov chain-based framework used to predict infrequent continuous loss due to signal fading, transient failure or stream burst. The interested reader, looking for the detail information of Markov loss model, should consult a more thorough treatise on the matter, such as [21].

## 4.2 Simulation results

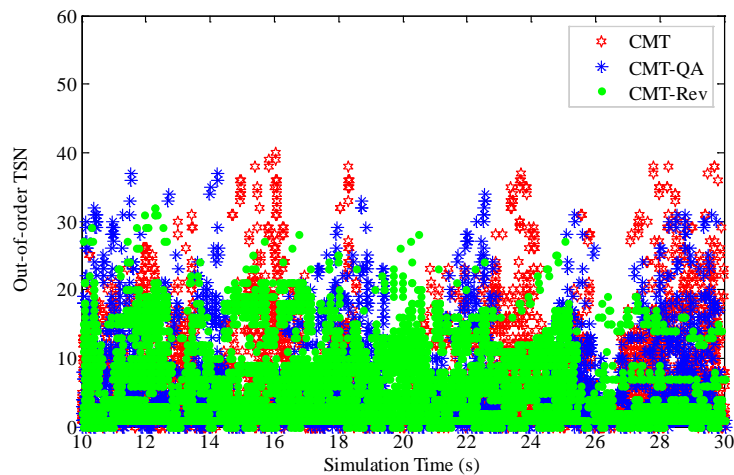
This subsection presents the performance evaluation and comparison between the classic CMT, CMT-QA solution [22], and the proposed CMT-Rev. To make it convenient, we illustrate the results of the classic CMT as ‘CMT’ in the test result figures, and the results with CMT-QA and CMT-Rev solution are illustrated as ‘CMT-QA’ and ‘CMT-Rev’, respectively.

1) *Sending and receiving TSN*: Fig. 7 portrays sending and arrival times of several SCTP data chunks when the classic CMT, CMT-QA, and CMT-Rev are used, respectively. In the classic CMT, the sender uses the round robin strategy to schedule SCTP data chunks over all the available paths equally, without considering the path quality differences. CMT-QA scheme senses the transmission efficiency for each path and provides a path handling capacity-aware data distribution strategy. However, it splits SCTP packets over all available paths without considering the fact that the path with unfavorable conditions may degrade the whole performance of data delivery. With the receiver-cooperative sending rate-aware data distribution strategy, CMT-Rev can predict path transmission efficiency and decide the sending path based on it. Moreover, it provides an adaptive ‘CMT-to-Partial CMT’ adjustment strategy, when severe receive buffer blocking is detected, CMT-Rev will disable the path with unfavorable transmission condition for data delivery, conversely, it uses all available paths for parallel transmission and bandwidth aggregation. In this way, CMT-Rev can avoid the need for most reordering and it achieves higher sending and receiving TSN than both the classic CMT and CMT-QA schemes.



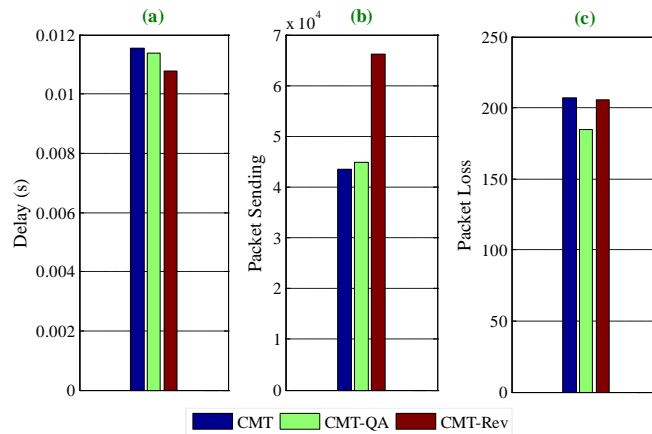
**Fig. 7.** Comparison of sending and receiving time of packets

2) *Out-of-order packets*: The out-of-order data reception at the receiver will incur additional packet reordering and recovery time [31]. In CMT-QA solution [22], the out-of-order TSN ( $O^3$ -TSN), obtained by the offset between the TSNs of two consecutively received data chunks, is used to reflect the characteristics of CMT-based data delivery over heterogeneous wireless network environment. We here employ the  $O^3$ -TSN metric to compare the performance among the classic CMT, CMT-QA, and our proposed CMT-Rev solution. In order to better illustrate the comparison, the results between  $t=10s$  and  $t=30s$  are presented (part of congestion avoidance phase), representative for the whole simulation results. As Fig. 8 shows, both the classic CMT and CMT-QA generate more out-of-order chunks and require increased packet reordering than the CMT-Rev solution. CMT-Rev takes into consideration the sending rate of each path, which is advertised by the receiver. It selects suitable candidate paths that have high transmission efficiency for concurrent multipath data transfer. In this way, CMT-Rev reduces the out-of-order data arrival and consequently performs better than the other two solutions. When comparing the three schemes, it is noted that peak out-of-order data reception at the receiver is approximately 41 using the classic CMT and about 38 using CMT-QA, while it is only close to 30 when using the CMT-Rev solution.

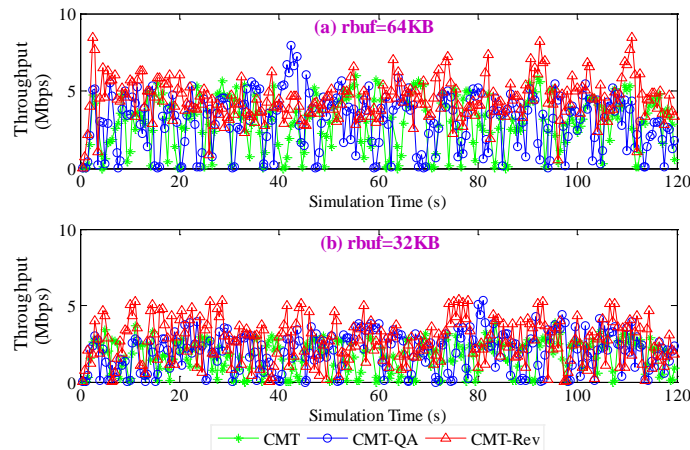


**Fig. 8.** Comparison of out-of-order TSN

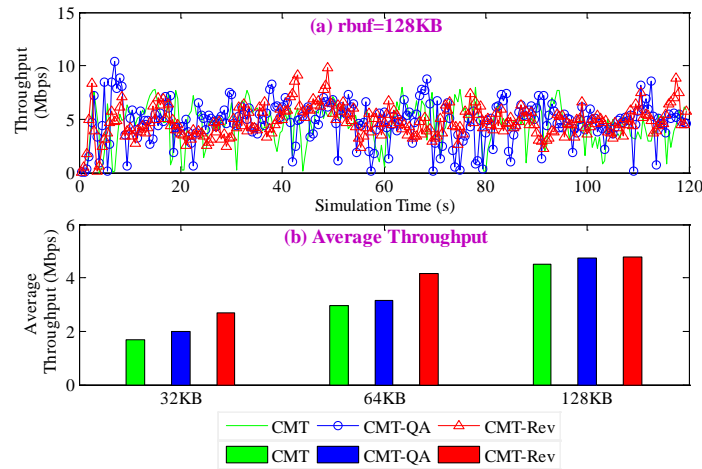
3) *Packet delay and loss*: **Fig. 9** (a) shows the comparison of packet delay when using the classic CMT, CMT-QA, and the proposed CMT-Rev, respectively. **Fig. 9** (b) and (c) present the comparison of packet sending and receiving when the three schemes are employed. In terms of e2e delay, CMT-Rev performs 6.62% and 5.28% lower than the classic CMT and CMT-QA, respectively. This is because CMT-Rev takes end-to-end delay variance into account during path sending rate evaluation and disables the paths with low-quality (i.e. high delay) for data transmission, it correspondingly reduces the end-to-end packet delay. Since higher packet delay determines more data cannot be received and handed over to application layer in time. Therefore, compared with the classic CMT and CMT-QA scheme, CMT-Rev can achieve better users' quality of experience (QoE) for data delivery service. As for packet loss performance, when using the classic CMT, the packet loss rate is about 0.476% (the number of packets sent is 43462 and loss is 207), and about 0.411% when using CMT-QA (the number of packets sent is 44970 and loss is 185), while it is only 0.312% when using CMT-Rev (the number of packets sent is 66110 and loss is 206). Lower packet loss probability means less out-of-order data delivery, as well as less data chunk retransmissions. Hence, CMT-Rev performs the best to avoid the need for most reordering and retransmission among the three solutions compared.



**Fig. 9.** Comparison of packet delay and loss



**Fig. 10.** Comparison of throughput when using rbuf=64KB and 32KB



**Fig. 11.** Comparison of  $rbuf = 128KB$  and average throughput

4) *Average throughput*: Since the  $rbuf$  values used in the existing operating systems (OSs) varied from 32 KB to 64 KB and beyond, we compare the average throughput when sending data with  $rbuf$  sizes of 32 KB, 64KB and 128 KB, which are presented in Fig. 10 and Fig. 11, respectively. Three groups of simulation (figures 10 (a), (b), and 11 (a)) were presented in order to clarify the effect of the receiver buffer size on the throughput performance. It can be obviously seen from the three figures that the throughput of all solutions increases with the increase in receiver buffer size. Moreover, in both the classic CMT and CMT-QA solutions, the sender uses all the paths to transmit data chunks, without considering the fact that path dissimilarity is bound to serious out-of-order data arrival. A large number of out-of-order data chunks buffered at receiver will constrain the sender from sending any new data chunk and consequently decrease the throughput performance sharply. In contrast, the proposed CMT-Rev achieves better throughput performance than the classic CMT and CMT-QA. That is because CMT-Rev intelligently selects a subset of suitable paths for bandwidth aggregation and adaptively assigns them appropriate data flows, enabled by the receiver-cooperative path aggregation strategy and receiver-aware data scheduling algorithm. These factors help CMT-Rev to alleviate packet reordering and improve the throughput performance. As Figure 11 (b) shows, after 120s of simulation time with a 64KB receiver buffer, CMT-Rev's throughput is 41.57% higher than that of CMT and 32.91% higher than that of CMT-QA. With a 32KB receiver buffer size, the corresponding comparison of average throughput performance is 59.02% and 35.58% in favor of the proposed CMT-Rev solution, respectively. Similarly, CMT-Rev performs 5.24% and 0.57% better than the classic CMT and CMT-QA, respectively, when a 128KB receiver buffer is employed.

## 5. Conclusion

This paper presents a novel CMT-Rev solution, an extension for CMT that runs some important function at receiver necessitating the following aims: (i) load sharing between the sender and receiver, (ii) mitigating CMT's buffer blocking, and (iii) improving the CMT performance. CMT-Rev provides a newly receiver-driven path quality estimation mechanism to accurately determine each path's sending rate at receiver. It designs an adaptive receiver-cooperative path aggregation strategy to assemble a subset of suitable paths for parallel transmission and bandwidth aggregation. Moreover, CMT-Rev introduces an

innovative receiver-aware data scheduling algorithm to reduce buffer blocking problems and improve data delivery performance. Simulation results reveal that the proposed CMT-Rev solution outperformed existing CMT protocols in terms of transmission performance and quality of service in heterogeneous wireless network environment. We notice that energy consumption becomes a pressing concern in wireless networking [32-34]. Future work will focus on designing a receiver-driven energy-aware CMT solution in order to improve the CMT performance while reducing energy consumption.

## References

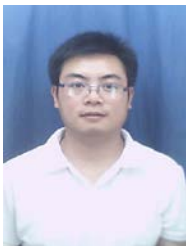
- [1] J. Wu, B. Cheng, C. Yuen, Y. Shang and J. Chen, "Distortion-Aware Concurrent Multipath Transfer for Mobile Video Streaming in Heterogeneous Wireless Networks," *IEEE Transactions on Mobile Computing*, vol. 14, no. 4, pp. 688-701, 2015. [Article \(CrossRef Link\)](#).
- [2] J. Wang, J. Liao, J. Wang, T. Li and Q. Qi, "Estimating the Effects of Multipath Selection on Concurrent Multipath Transfer," *KSI Transactions on Internet and Information Systems*, vol. 8, pp. 1406-1423, 2014. [Article \(CrossRef Link\)](#).
- [3] R. Stewart, "Stream Control Transmission Protocol," *IETF RFC 4960* (Proposed Standard), Sep, 2007. [Article \(CrossRef Link\)](#).
- [4] Ł. Budzisz, J. Garcia, A. Brunstrom and R. Ferrús, "A taxonomy and survey of SCTP research," *ACM Computing Surveys*, vol. 44, no. 4, pp. 1-36, Aug, 2012. [Article \(CrossRef Link\)](#).
- [5] Y. Cao, C. Xu, J. Guan and H. Zhang, "Qos-driven SCTP-based multimedia delivery over heterogeneous wireless networks," *SCIENCE CHINA Information Sciences*, vol. 57, no. 10, pp. 1-10, 2014. [Article \(CrossRef Link\)](#).
- [6] J. R. Iyengar, P. Amer and R. Stewart, "Concurrent Multipath Transfer Using SCTP Multihoming over Independent End-to-end Paths," *IEEE/ACM Transactions on Networking*, vol. 14, no. 5, pp. 951-964, Oct, 2006. [Article \(CrossRef Link\)](#).
- [7] P. Amer, M. Becke, T. Dreibholz, N. Ekiz, J. Iyengar, P. Natarajan, R. Stewart and M. Tuexen, "Load Sharing for the Stream Control Transmission Protocol (SCTP)," *draft-tuexen-tsvwg-sctp-multipath-08.txt*, 2014. [Article \(CrossRef Link\)](#).
- [8] C. Huang and M. Lin, "Fast Retransmission for Concurrent Multipath Transfer (CMT) over Vehicular Networks," *IEEE Communications Letters*, vol. 15, no. 4, pp. 386-388, 2011. [Article \(CrossRef Link\)](#).
- [9] C. Xu, E. Fallon, Q. Yuansong, L. Zhong and G. Muntean, "Performance Evaluation of Multimedia Content Distribution Over Multi-Homed Wireless Networks," *IEEE Transactions on Broadcasting*, vol. 57, no. 2, pp. 204-215, June, 2011. [Article \(CrossRef Link\)](#).
- [10] C.-M. Huang and M.-S. Lin, "Multimedia streaming using partially reliable concurrent multipath transfer for multihomed networks," *IET Communications*, vol. 5, no. 5, pp. 587-597, April, 2011. [Article \(CrossRef Link\)](#).
- [11] J. Wu, C. Yuen, B. Cheng, Y. Shang and J. Chen, "Goodput-Aware Load Distribution for Real-time Traffic over Multipath Networks," *IEEE Transactions on Parallel and Distributed Systems*, no. 99, 2014. [Article \(CrossRef Link\)](#).
- [12] C. Xu, Z. Li, J. Li, H. Zhang and G. Muntean, "Cross-layer Fairness-driven Concurrent Multipath Video Delivery over Heterogeneous Wireless Networks," *IEEE Transactions on Circuits and Systems for Video Technology*, no. 99, 2015. [Article \(CrossRef Link\)](#).
- [13] Y. Cao, C. Xu, J. Guan and H. Zhang, "Receiver-driven SCTP-based multimedia streaming services in heterogeneous wireless networks," in *Proc. of IEEE Int. Conf. on Multimedia and Expo*, pp. 1-6, Jul, 2014. [Article \(CrossRef Link\)](#).
- [14] J. Minho, T. Maksymyuk, R.L. Batista, T.F. Maciel, A.L.F. Almeida and M. Klymash, "A Survey of Converging Solutions for Heterogeneous Mobile," *IEEE Wireless Communications*, vol. 21, no. 8, pp. 54-62, 2014. [Article \(CrossRef Link\)](#).
- [15] I. Rhee, V. Ozdemir and Y. Yi, "TEAR: TCP Emulation at Receivers-Flow Control for Multimedia Streaming," *Technical Report*. [Article \(CrossRef Link\)](#).
- [16] T. D. Wallace and A. Shami, "A review of multihoming issues using the stream control transmission protocol," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 2, pp. 565-578, 2012. [Article \(CrossRef Link\)](#).
- [17] T. Dreibholz, E. P. Rathgeb, I. Rungeler, R. Seggelmann, M. Tuxen and R. Stewart, "Stream Control Transmission Protocol: Past, Current and Future Standardization Activities," *IEEE Communications Magazine*, vol. 49, no. 4, pp. 82-88, Apr, 2011. [Article \(CrossRef Link\)](#).



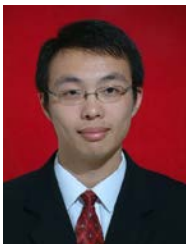
- [18] C. Huang and M. Lin, "Fast Retransmission for Concurrent Multipath Transfer (CMT) over Vehicular Networks," *IEEE Communications Letters*, vol. 15, no. 4, pp. 386-388, 2011. [Article \(CrossRef Link\)](#).
- [19] B. Wang, W. Feng, S. Zhang and H. Zhang, "Concurrent Multipath Transfer Protocol Used in Ad Hoc Networks," *IET Communications*, vol. 4, no. 7, pp. 884-893, Apr, 2010. [Article \(CrossRef Link\)](#).
- [20] J. Baek, P. Fisher, M. Jo and H. Chen, "A Lightweight SCTP for Partially Reliable Overlay Video Multicast Service for Mobile Terminals," *IEEE Transactions on Multimedia*, vol. 12, no. 7, pp. 754-766, Nov, 2010. [Article \(CrossRef Link\)](#).
- [21] T.D. Wallace and A. Shami, "Concurrent Multipath Transfer Using SCTP: Modelling and Congestion Window Management," *IEEE Transactions on Mobile Computing*, vol. 13, no. 11, pp. 2510-2523, Feb, 2014. [Article \(CrossRef Link\)](#).
- [22] C. Xu, T. Liu, J. Guan, H. Zhang and G. Muntean, "CMT-QA: Quality-aware Adaptive Concurrent Multipath Data Transfer in Heterogeneous Wireless Networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 11, pp. 2193-2205, Nov, 2013. [Article \(CrossRef Link\)](#).
- [23] Y. Cao, C. Xu, J. Guan and H. Zhang, "CMT-CC: Cross-Layer Cognitive CMT for Efficient Multimedia Distribution over Multi-homed Wireless Networks," *Wireless Personal Communications*, vol. 82, no. 3, pp. 1643-1663, 2015. [Article \(CrossRef Link\)](#).
- [24] F. Perotto, C. Casetti and G. Galante, "SCTP-based Transport Protocols for Concurrent Multipath Transfer," in *Proc. of IEEE Wireless Communications and Networking Conference*, Mar, 2007. [Article \(CrossRef Link\)](#).
- [25] M. Becke, T. Dreibholz, H. Adhari and E.P. Rathgeb, "On the fairness of transport protocols in a multi-path environment," in *Proc. of IEEE Int. Conf. on Communications*, pp. 2666-2672, 2012. [Article \(CrossRef Link\)](#).
- [26] T. Dreibholz, M. Becke, J. Pulinthanath and E. Rathgeb, "Applying TCP-Friendly Congestion Control to Concurrent Multipath Transfer," in *Proc. of IEEE Int. Conf. on Advanced Information Networking and Applications*, pp. 312-319, 2010. [Article \(CrossRef Link\)](#).
- [27] J. Liao, J. Wang and X. Zhu, "cmpSCTP: An Extension of SCTP to Support Concurrent Multi-Path Transfer," in *Proc. of IEEE Int. Conf. on Communications*, pp. 5762-5766, May, 2008. [Article \(CrossRef Link\)](#).
- [28] D. Chiu and R. Jain, "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks," *Computer Networks and ISDN Systems*, vol. 17, no. 1, pp. 1-14, Jun, 1989. [Article \(CrossRef Link\)](#).
- [29] S. Shailendra, R. Bhattacharjee and Sanjay K. Bose, "MPSCTP: A Simple and Efficient Multipath Algorithm for SCTP," *IEEE Communications Letters*, vol. 15, no. 10, pp. 1139-1141, Oct, 2011. [Article \(CrossRef Link\)](#).
- [30] UC Berkeley, LBL, USC/ISI and Xerox Parc, NS-2 documentation and software, version 2.35. [Article \(CrossRef Link\)](#).
- [31] J. Wu, J. Yang, Y. Shang, B. Cheng and J. Chen, "SPMLD: Sub-Packet based Multipath Load Distribution for Real-time Multimedia Traffic," *Journal of Communications and Networks*, vol. 16, no. 5, pp. 548-558, 2014. [Article \(CrossRef Link\)](#).
- [32] Y. Zhang and Q. Li, "Exploiting ZigBee in Reducing WiFi Power Consumption for Mobile Devices," *IEEE Transactions on Mobile Computing*, vol. 13, no. 12, pp. 2806-2819, 2014. [Article \(CrossRef Link\)](#).
- [33] B. Lakshminarayanan and M. Krishanan, "Avoiding Energy Holes Problem using Load Balancing Approach in Wireless Sensor Network," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 5, pp. 1618-1637, May, 2014. [Article \(CrossRef Link\)](#).
- [34] H. Shen, S. Lv, X. Wang and X. Zhou, "Energy-efficient Routing in MIMO-based Mobile Ad hoc Networks with Multiplexing and Diversity Gains," *KSII Transactions on Internet and Information Systems*, vol. 9, no. 2, pp. 700-713, Feb, 2015. [Article \(CrossRef Link\)](#).



**Yuanlong Cao** received his Ph.D. degree from Institute of Network Technology, Beijing University of Posts and Telecommunications (BUPT) in Sep 2014, received his MS degree in software engineering from BUPT in July 2008, and received his BS degree in computer science and technology from Nanchang University, China, in July 2006. He worked as an intern/engineer in BEA TTC, IBM CDL, and DT Research (Beijing) from 2007 to 2011, respectively. He is a Lecturer with the School of Software at Jiangxi Normal University, China. His research interests include multimedia communications and next generation Internet technology.



**Qinhua Liu** is now working as an experimentalist in the Software of School, JiangXi Normal University (JXNU). He received the B.S. degree in software engineering from JXNU, China, in 2007, and his Master degree in management science and engineering in 2011, from JXNU. His research interests include multimedia networking and next generation Internet technology.



**Yi Zuo** is now working as a project manager in the Innovation Fund for Small Technology-based Firms (STF) of Jiangxi province. He received the B.S. degree from Jiangxi University of Finance and Economics, China, in July 2007. He received the Master Degree from Technische Universität Dresden, German, in May 2009. His research interests include next generation service creation technology.



**Minghe Huang** is a professor of the Software of School at JiangXi Normal University (JXNU), and the Jiangxi provincial government counselor. He has published more than 50 research papers in the areas of computer networks, communications, and information theory. He was the chairman of Software of School at JXNU, and the standing director of the Association of Fundamental Computing Education in Chinese Universities (AFCEC). Prof. Huang was elected as a “University Young and Middle-aged Academic Leaders in Jiangxi Province”, and “Distinguished Teacher of Jiangxi Province”.