

RESEARCH

Open Access



Receiver-assisted cellular/wifi handover management for efficient multipath multimedia delivery in heterogeneous wireless networks

Yuanlong Cao^{1*}, Qinghua Liu¹, Yi Zuo², Guoliang Luo¹, Hao Wang¹ and Minghe Huang¹

Abstract

In a heterogeneous wireless environment, mobility particularly vertical handover between WiFi and cellular data service has to be supported. The support for multihoming feature makes Stream Control Transmission Protocol (SCTP) become a promising transport layer-based handover scheme for end users. Although the benefits of applying SCTP for multimedia delivery and handover have been proved to be extremely useful, there are still many challenges that remain: (i) current sender-centric SCTP handover management solutions fail to balance the computational overhead between a sender and receiver and (ii) they also suffer from performance degradation due to the rigid three-duplicated-SACK-based loss detection and recovery strategy. This paper proposes SCTP-Rev⁺, a new receiver-assisted cellular/WiFi handover management mechanism for SCTP-based multimedia transport aiming to (i) optimize SCTP handover management and achieve overhead balancing between the sender and receiver, by moving path estimation and handover decision operations from the sender onto receiver and (ii) improve SCTP loss recovery ability and multimedia delivery performance, by providing SCTP with a simple but effective retransmission-aware loss detection/recovery mechanism. We show that the proposed solution outperforms the current SCTP schemes in terms of goodput performance and multimedia delivery quality.

Keywords: Multipath transport protocol, SCTP, Handover management, Multimedia communication

1 Introduction

Recently, significant advancements in wireless access systems, such as wireless local area networks (802.11 WiFi family) and cellular networks (UMTS, WiMAX, etc.), offer a mobile user ubiquitous Internet access [1]. In such an environment, mobility particularly vertical handover between the heterogeneous access networks has to be supported [2]. Meanwhile, promoted by the latest WiFi and cellular technological progress, wireless terminals (i.e., mobile PC, smartphones, etc.) are embedded with multinet interface and enabled simultaneous access to multiple heterogeneous networks [3, 4]. Such multihomed devices can improve their throughput performance and enhance user's quality of experience (QoE)

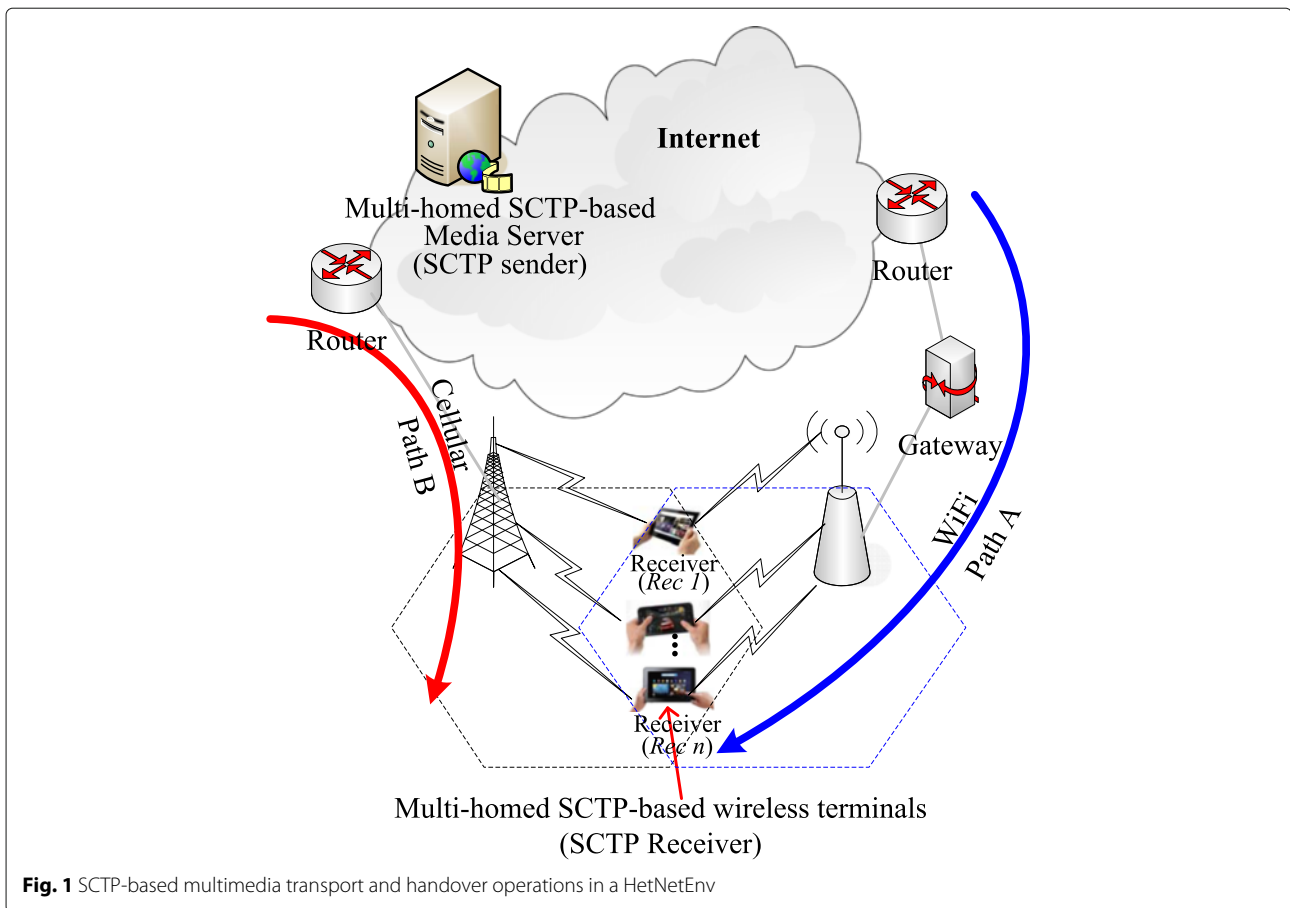
by making use of multiple links either simultaneously or alternatively in a heterogeneous wireless network, supported by the Stream Control Transport Protocol (SCTP) [5]. Due to its multihoming and partial reliability features [6], the SCTP has been considered as a good transport-layer technology suitable for continuously real-time multimedia delivery and seamless handover across a heterogeneous network environment (HetNetEnv).

Figure 1 illustrates the usage of SCTP-based multimedia transport and handover in a HetNetEnv. As the figure shows, a SCTP-based media server (a sender) is communicated with several receivers (Rec 1, Rec 2, ..., Rec *n*) by using two paths, which are path *A* that is a primary path and path *B* that is a secondary path. In such a scenario, the SCTP sender manages the two paths by making use of three major functions: (i) using path *A* and path *B* for multimedia content delivery and fault tolerance, respectively; (ii) using selective acknowledgment (SACK)

*Correspondence: ylcao@jxnu.edu.cn

¹Jiangxi Normal University, 330022 Nanchang, China

Full list of author information is available at the end of the article



and Heartbeat (HB) chunks to detect the connectivity of path *A* and path *B*, respectively [7]; and (iii) *handover* which means using path *B* as a new primary path for multimedia transmission once the path *A* breaks (the maximum number of consecutive timeouts on path *A* is reached to the value of Path_Max_Retrans). Such features make SCTP become a preferred transport-layer protocol for content-rich multimedia transport and handover in a HetNetEnv [8, 9].

Although the benefits of using SCTP for multimedia transport and handover in a HetNetEnv have been proved to be useful, there are still many remaining issues and concerns on this topic. The first concern when employing SCTP for multimedia streaming is related to SCTP's TCP-like AIMD-based congestion control mechanism, which makes SCTP (i) cut the congestion window (*cwnd*) down to half for a single loss and (ii) enter slow-start phase when primary path switching occurs [10]. Such a "rigid" congestion control mechanism will cause bursty and frequent transmission rate fluctuations, and undoubtedly degrade users' quality of experience (QoE) for multimedia streaming services in a HetNetEnv, in which packet loss and handover tend to occur abruptly due to the characteristics of heterogeneous wireless systems [11].

The second concern when applying SCTP to multimedia delivery is the load-balancing problem between the sender and receiver. Current SCTP extensions [12–19] and multimedia-related solutions [1, 4, 20–22] depend solely upon the sender to run flow/congestion control and handover management operations. However, taking Fig. 1 as an example, the sender may become a "processing bottleneck" in the system when very large amounts of receivers with multiple paths are communicated. In fact, running some operations (i.e., flow and congestion control) at receiver is a ideal solution to alleviate the workload of the sender [23, 24]. Moreover, such a receiver-controlled solution makes the receiver exploit the first-hand network information obtained in its forward path for flow control decision.

The third concern of the SCTP-based multimedia wireless transmission is the "rigid" three-duplicated-SACK-based fast retransmission mechanism. In SCTP, a lost chunk cannot be retransmitted until its *transmission sequence number* (TSN) is reported as missing three times successively [25]. As a fully reliable and fully ordered transport protocol, the SCTP does not hand over the *out-of-order* (O^3) packets to upper application layer until a lost packet is received successfully, that is, a packet

loss may result in numerous O^3 packets buffered in the space-limited receiver buffer and cause a severe hand over delay problem. Worse still for the time-sensitive multimedia streaming services, the delayed delivery of multimedia packets may lead to some of these received packets becoming unplayable.

In order to address above concerns, this paper *extends our previous work Sctp-Rev* [26] to propose a receiver-assisted cellular/WiFi handover management mechanism (dubbed as Sctp-Rev⁺) for Sctp-based heterogeneous multimedia delivery. Sctp-Rev⁺ moves some operations including path sending rate estimation and primary path selection from the sender onto receiver in order to achieve load-balancing between the sender and receiver. Moreover, Sctp-Rev⁺ includes a simple retransmission-aware model that makes it possible to handle data loss and trigger proper fast retransmission. The major contributions of the paper are introduced as follows:

- It moves the path sending rate control and primary path selection operations from the sender onto the receiver side in order to possibly help Sctp alleviate the workload of the sender.
- It develops a receiver-triggered path fast recovery strategy in order to prevent bursty packet injection while maximizing resource utilization when a primary path switching event occurs.
- It includes an efficient loss detection/recovery mechanism in order to make a Sctp sender possibly identify a packet loss event timely and improve the performance of multimedia transmission.

The remainder of this paper is organized as follows. Section 2 introduces the detail design of the proposed Sctp-Rev⁺ solution. Section 3 presents the performance evaluation and analysis of the Sctp-Rev⁺ solution. Section 4 concludes the paper and gives some interesting directions for future work.

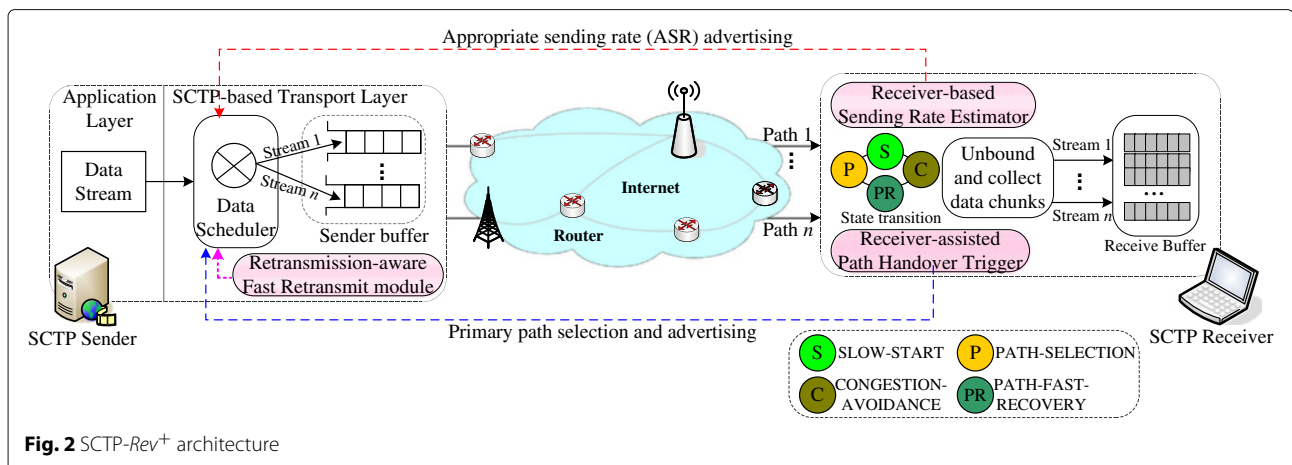
2 Sctp-Rev⁺ detail design

In current Sctp handover solutions, the sender needs to monitor per-path's transmission quality and make a handover decision. In contrast, the Sctp-Rev⁺ sender does not need to detect the transmission quality for each path, it just uses per-path's sending rate, which is calculated and advertised by the receiver, as the transmission quality metric to make a handover decision. Such way possibly helps the Sctp-Rev⁺ to reduce the sender's computing overhead and achieve a desired load-sharing between the sender and receiver.

Figure 2 shows the architecture of Sctp-Rev⁺ system. Sctp-Rev⁺ mainly consists of a *receiver-based sending rate estimator* (SRE-*rev*), a *receiver-assisted path handover trigger* (PST-*rev*), and a *retransmission-aware fast retransmit* (RAF-*rtx*) module. The major functions of the three components are described as follows:

- SRE-*rev* is devoted to calculating per-path's sending rate at Sctp receiver side and giving the sender feedback on the paths' sending rate value.
- PST-*rev* is devoted to choosing a proper primary path at Sctp receiver and giving the sender feedback on the selected primary path information for handover decision.
- RAF-*rtx* is devoted to helping the Sctp sender possibly detect packet loss timely and trigger fast retransmission for loss recovery.

Like TEAR [24], a well-known receiver-driven TCP-based flow control solution, Sctp-Rev⁺ runs seven states at receiver side for flow control, which are Slow Start, Congestion Avoidance, Slow Start Ready, Timeout, Congestion Avoidance Ready, and Fast Recovery. The state transition among the seven states has been detailed in the TEAR solution. Meanwhile, the Sctp-Rev⁺ uses the *round* information, which is already defined in the TEAR solution, to measure the values of *round-trip time* (RTT)



and retransmission timeout (RTO). For convenience, we introduce the usage of round and the round-based RTT and RTO calculations. Next, the major design of SRE-*rev*, PST-*rev*, and RAF-*rtx* are detailed, respectively.

Definition 1 A round begins when: (i) a selective acknowledgment (SACK) chunk with the path sending rate (PSR) values is received by the sender and (ii) the advertised PSR values are used by the sender for packet delivery.

Definition 2 A round comes to end and a new round is starting once a new SACK is received by sender and the carried PSR values is used for each path's sending rate adjustment.

In SCTP-*Rev*⁺, each packet required to be sent will be labeled with a round ID (rID) by the sender in order to tell the receiver about the round info. Meanwhile, the SCTP-*Rev*⁺ receiver will record a timestamp for each SACK chunk that is to be transmitted. Based on the SACK delivery time and rID information, the SCTP-*Rev*⁺ receiver runs the RTT and RTO calculations following the steps below:

- 1) Once having a SACK S_i to be sent, the receiver records the sending time of S_i (denoted as t_i).
- 2) When the sender receives S_i , it picks up the advertised PSR information for packet delivery, this means a new round round_j begins.
- 3) When the first packet with the round_j information, $p_{\text{round}_j}^0$, arrives at the receiver side, the receiver records the receiving time of $p_{\text{round}_j}^0$ (denoted as t_{i+1}).
- 4) The receiver calculates the RTT value by using the following equation:

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

where RTT_{cur} is the current RTT value measured on each path and α is coefficient factor with a default value of $\frac{1}{8}$.

- 5) The RTO calculation in the SCTP-*Rev*⁺ solution is the same as that in the regular SCTP, which can be computed by

$$\text{RTO} = \text{SRTT} + 4 \times \text{RTTVAR}, \quad (2)$$

where SRTT and RTTVAR can be initialized by $(t_{i+1} - t_i)$ and $\frac{t_{i+1} - t_i}{2}$, respectively.

2.1 Receiver-based sending rate estimator (SRE-*rev*)

As mentioned above, the standard SCTP utilizes a TCP-like AIMD-based sending rate and congestion control. Suppose there are ψ paths (d_1, d_2, \dots, d_ψ) within an SCTP association, when a packet is scheduled on path

d_δ ($d_\delta \in (d_1, d_2, \dots, d_\psi)$), the AIMD-style mechanism [27] used in SCTP for d_δ 's sending rate control can be generally expressed by

$$\text{rate}_{i+1}^{d_\delta} = \begin{cases} \text{rate}_i^{d_\delta} + \Delta\text{rate}, & \text{if no congestion is detected;} \\ \frac{1}{2} \times \text{rate}_i^{d_\delta}; & \text{otherwise.} \end{cases} \quad (3)$$

Due to the asymptotic behavior of AIMD algorithm, the standard SCTP in congestion avoidance phase will increase the sending rate of d_δ by Δrate after each one RTT period (let $\text{packet}_{\text{size}}$ refers to the packet size, $\Delta\text{rate} \approx \frac{\text{packet}_{\text{size}}}{\text{RTT}}$). In other words, let us suppose that there are M packets required to be sent within one RTT, the sending rate of d_δ will be increased by Δrate ($\Delta\text{rate} \approx \frac{1}{M} \times \frac{\text{packet}_{\text{size}}}{\text{RTT}}$) when each of the transmitted packets is received and acknowledged by the receiver, and it will be halved if a congestion condition is encountered.

Since the AIMD-style rate control mechanism is bound to the abrupt and frequent sending rate fluctuations, it cannot make SCTP enable smooth high-quality service for multimedia streaming applications in a varying wireless environment. As a remedy, we introduce a receiver-based sending rate estimator (SRE-*rev*) necessitating the following aims: (i) running a weighted moving average (WMA)-based AIMD rate control mechanism at receiver to determine the sending rate for each path and (ii) alleviating the sender's computational load and possibly reducing bursty rate fluctuations. With the WMA-based AIMD rate control mechanism, the sending rate calculation for d_ℓ ($d_\ell \in (d_1, d_2, \dots, d_\psi)$) can be expressed by

$$\text{rate}_{i+1}^{d_\ell} = \begin{cases} \Xi \times \text{rate}_i^{d_\ell} + (1 - \Xi) \times \frac{L_{\text{size}}}{N \times \text{RTT}}, & \text{if PLR} < \text{thresh;} \\ (1 - \Phi) \times \text{rate}_i^{d_\ell} + \Phi \times \frac{L_{\text{size}}}{N \times \text{RTT}}, & \text{otherwise,} \end{cases} \quad (4)$$

where N is the number of packet (the DATA chunks on primary path or the HB chunks on secondary path) received by receiver within one round. L_{size} is the SCTP packet (DATA or HB) size. Ξ and Φ are weighting factors with default values of $\frac{1}{2}$ for fairness. PLR and thresh are the loss rate and predefined loss threshold, respectively.

Compared to the classic AIMD-based rate control mechanism, it is found that (i) in case of a network congestion, we have $\text{rate}_{i+1}^{d_\ell} \approx \frac{1}{2} \times \text{rate}_i^{d_\ell}$ since $\Delta\text{rate} = \frac{L_{\text{size}}}{N \times \text{RTT}} \approx 0$. In this case, SRE-*rev* refers to the regular AIMD operations where additive-increase/multiplicative-decrease is used for sending rate control. This feature makes SCTP-*Rev*⁺ perform a same performance as SCTP in terms of TCP friendliness in case of congestion. (ii) Otherwise, we can suppose that we have $\text{rate}_i^{d_\ell} \approx \Delta\text{rate}$ when the gap between $\text{rate}_i^{d_\ell}$ and Δrate is less than a given value. Therefore, SRE-*rev* can cut the sending rate

down to a proper value for network congestion alleviation or speed the sending rate up to a maximum value for bandwidth utilization improvement. These features make SCTP-Rev⁺ adaptively utilize the wireless resource.

Before giving the sender feedback on the estimated rate_{i+1}^{d_ℓ} value, the SCTP-Rev⁺ receiver will smooth the estimated sending rate by using the following equation:

$$\text{Rate}_e^{d_\ell} = \frac{\Upsilon \cdot \Xi_g}{\sum_{s=1}^{\kappa} \Upsilon_s}, \quad (5)$$

where Ξ_g is a matrix used for storing per-path's κ most recent sending rate values, which can be expressed as below using Eq. (6):

$$\Xi_g = [\text{rate}_{g,1}, \text{rate}_{g,2}, \dots, \text{rate}_{g,\kappa}]. \quad (6)$$

Meanwhile, the coefficient vector Υ can be defined as in Eq. (7),

$$\Upsilon = [\Upsilon_1, \dots, \Upsilon_s, \dots, \Upsilon_\kappa], \quad (7)$$

where Υ_s ($s \in [1, \kappa]$) is calculated as in Eq. (8),

$$\Upsilon_s = \begin{cases} 1, & \text{if } 1 \leq s \leq \frac{\kappa}{2}; \\ 1 - \frac{s - \frac{\kappa}{2}}{\frac{\kappa}{2} + 1}, & \text{if } \frac{\kappa}{2} \leq s \leq \kappa, \end{cases} \quad (8)$$

In SCTP-Rev⁺, when the current sending rate the sender used is unexpected or a timeout event is detected, the receiver will immediately give the sender feedback on the estimated sending rate $\text{Rate}_e^{d_\ell}$, by using a modified SACK chunk (for a primary path) or a modified HB ACK chunk (for a secondary path), as shown in Fig. 3a, b, respectively. Both of the extended SACK and HB ACK chunks include a new 64-bit *Advertised Receiver Desired Sending Rate* for the sending rate $\text{Rate}_e^{d_\ell}$ advertisement. Moreover, a delayed response timer of one RTT will be launched at the SCTP-Rev⁺ receiver when a new round is starting. This way can help the receiver to capture a timeout event.

2.2 Receiver-assisted path handover trigger (PST-rev)

According to the proposed SCTP standard [5], an SCTP sender need to detect the primary path error count (denoted as $P^2_error_count$) and conduct primary path handover once the value of $P^2_error_count$ exceeds the maximum number of consecutive timeouts Path_Max_Retrans [28]. Such a sender-centric failover-based handover mechanism will inevitably degrade the SCTP performance due to (i) failure detection on the primary path is a time-consuming process (more than 60 s [11]) and (ii) frequent “ping-pong” primary path handover may increase the sender's load and overhead. Moreover, when SCTP switches to a new primary path, it may experience a slow-start phase on the new path because of a timeout event. This may cause a sending rate fluctuation and influence user-perceived quality of multimedia communications.

To resolve the aforesaid issues, this section introduces a *receiver-assisted path handover trigger* (PST-rev), which runs at the SCTP-Rev⁺ receiver with the goals of optimizing SCTP handover performance and minimizing the handover overhead of sender. To this end, the PST-rev compares the sending rate of each path (period of 1 round) and select a path with the highest sending rate as the primary path for handover operations. Let Rate_e , d_{cp} , and d_{ap} be the path sending rate calculated by SRE-rev, the current primary path, and a secondary path with the highest Rate_e value within the secondary path list, respectively. Let $\text{Rate}_e^{d_{cp}}$ and $\text{Rate}_e^{d_{ap}}$ be the estimated sending rate on d_{cp} and d_{ap} , respectively. Once having $\text{Rate}_e^{d_{ap}} > \text{Rate}_e^{d_{cp}}$, the path d_{ap} will be chosen and advertised as the new primary path by the receiver. Before advertising the d_{ap} information to the sender, PST-rev will calculate an optimal sending rate (denoted as $\text{Rate}_e^{\text{new}}$) for d_{ap} , by using the following equation:

$$\text{Rate}_e^{\text{new}} = \text{Rate}_e^{d_{cp}} + \text{sgn}(\Delta) \times \text{Rate}_e^{d_{ap}} \times \frac{\text{RTT}_{\min}}{\text{RTT}_{\text{curr}}}, \quad (9)$$

where RTT_{curr} and RTT_{\min} are the current RTT measured on d_{cp} and the minimum RTT measured on d_{ap} ,

Type=3	Chunk Flags=0	Chunk Length=variable
Cumulative TSN Ack		
Advertised Receiver Window Credit (a_rwnd)		
Advertised Receiver Desired Sending Rate (a_rdsr)		
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		

a

Type=0x04	Chunk Flags=0	Chunk Length=variable
Heartbeat Info Type=1		Heartbeat Info Length = variable
Advertised Receiver Desired Sending Rate (a_rdsr)		

b

Type=X	Chunk Flags=0	Chunk Length=variable
Advertised Primary Path Path Identifier (a_pid)		
Advertised Receiver Desired Sending Rate (a_rdsr)		

c

Fig. 3 Formats of **a** SCTP SACK chunk extension, **b** SCTP HB chunk extension, and **c** primary path switching notification chunk

respectively. The purpose of RTT_{\min} is to avoid sporadic losses [29]. $\frac{RTT_{\min}}{RTT_{\text{curr}}}$ is a weighting factor. Suppose there are ψ paths (d_1, d_2, \dots, d_ψ) within the SCTP association, and let $RTT_{d_{cp}}$ and $RTT_{d_{ap}}$ be the estimated RTT on d_{cp} and d_{ap} , respectively. We have

$$Rate_e^{d_{ap}} = \underbrace{Rate_e^{d_1}, Rate_e^{d_2}, \dots, Rate_e^{d_\psi}}_{\max}, \quad (10)$$

$$\Delta = RTT_{d_{cp}} - RTT_{d_{ap}}, \quad (11)$$

$$\text{sgn}(\Delta) = \begin{cases} 1, & \text{if } \Delta > 0; \\ 0, & \text{if } \Delta = 0; \\ -1, & \text{if } \Delta < 0, \end{cases} \quad (12)$$

The usage of $Rate_e^{\text{new}}$ can help SCTP-Rev⁺ not only to avoid injecting traffic bursts into d_{ap} but also to maximize the network utilization of d_{ap} when SCTP-Rev⁺ switches to d_{ap} for multimedia content delivery. To tell the sender the selected primary path d_{ap} , the SCTP-Rev⁺ receiver will inform the sender of d_{ap} 's path identifier (*pid*), by using a *Primary Path Switching Notification* (PPSN) chunk, as shown in Fig. 3c. The details of *pid* has been introduced in [30]. In order to trigger a fast primary path handover, the SCTP-Rev⁺ receiver uses a reverse path with the lowest RTT for the PPSN feedback. When the PPSN chunk is received, the sender switches to the d_{ap} and uses the $Rate_e^{\text{new}}$ value as the sending rate for multimedia delivery.

To summarize, the SCTP-Rev⁺ runs the primary path selection and handover operations as follows: (i) the receiver uses the SRE-*rev* module to calculate each path's sending rate and then uses the sending rate as a metric to sort these paths in a descending order; (ii) the receiver uses the PST-*rev* module to compute a proper sending rate $Rate_e^{\text{new}}$ for the first path (i.e., d_{ap}) within the sorted list; (iii) the receiver gives the sending feedback on d_{ap} 's *pid* and $Rate_e^{\text{new}}$ value, by using the PPSN chunk; and (iv) the sender switches to the path d_{ap} and uses $Rate_e^{\text{new}}$ as the sending rate for data delivery. The pseudo code of SCTP-Rev⁺-based path selection and handover is presented in Algorithm 1; we assume that the SCTP association includes two paths, d_{cp} and d_{ap} , where d_{cp} is the primary path, and d_{ap} is the secondary path.

2.3 Retransmission-aware fast retransmit module (RAF-*rtx*)

In a lossy wireless network environment, the performance of SCTP is largely dependent upon its loss recovery mechanism. Taking Fig. 4 for example, due to SCTP's fully reliable and fully ordered nature, even some packets (i.e., TSN 1-2 and TSN 5-6) are received by the receiver, they cannot be handed over to the upper layer at once until the miss-

Algorithm 1 SCTP-Rev⁺-based path selection and handover operations

-
- At the receiver side:
- 1: calculating per-path sending rate by using Eqs. (4), (5), (6), (7) and (8);
 - 2: **setting** $Rate_e^{d_{cp}}$ = the sending rate of d_{cp} ;
 - 3: **setting** $Rate_e^{d_{ap}}$ = the sending rate of d_{ap} ;
 - 4: **if** $Rate_e^{d_{ap}} > Rate_e^{d_{cp}}$ **then**
 - 5: **if** d_{ap} is active **then**
 - 6: using Eq. (9) to compute a smoothed sending rate $Rate_e^{\text{new}}$ for d_{ap} ;
 - 7: giving the sender feedback on the d_{ap} 's *pid* and $Rate_e^{\text{new}}$ value;
 - 8: **else**
 - 9: continuing the PST-*rev* module;
 - 10: **end if**
 - 11: **end if**
- At the sender side:
- 12: switching to the d_{ap} and using $Rate_e^{\text{new}}$ as the ending rate for data delivery;
-

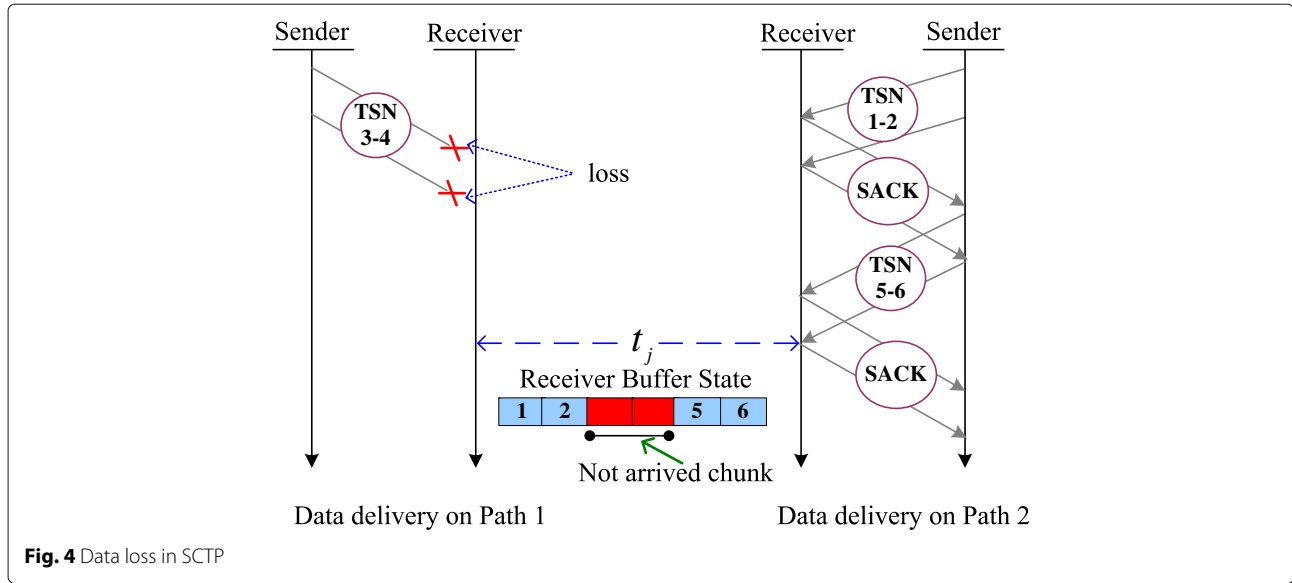
ing packets (i.e., TSN 3-4) are retransmitted and received. However, like it was previously mentioned, the standard SCTP does not mark a loss chunk for fast retransmission until its TSN is reported missing three times successively. Such a rigid three-duplicated-SACK-based packet loss detection/recovery mechanism will degrade user perception quality of real-time multimedia streaming services.

It would achieve desired performance if a lost packet can be detected timely and further be retransmitted rapidly by a sender [31]. Actually, the retransmission queue length is mostly used as an important parameter for loss detection and managing retransmissions [32, 33]. Motivated by these facts, this section concentrates on improving the performance of SCTP-based multimedia streaming by constructing a simple, yet effective loss detection and recovery model which is dubbed as RAF-*rtx*. RAF-*rtx* aims to make SCTP-Rev⁺ possible to identify the lost packets timely and enable loss recovery rapidly in a lossy wireless environment, by detecting the jitter indicator of the retransmission queue length.

Let d_{cp} be the primary path, and let $cwnd_{d_{cp}}$ and $Q(0 \leq Q \leq cwnd_{d_{cp}})$ refer to the *cwnd* size and the retransmission count of path d_{cp} , respectively. We estimate the jitter indicator of Q , Q^Δ , by using the following equation:

$$Q^\Delta = \frac{Q_{\text{curr}} - \bar{Q}}{Q_{\text{max}} - Q_{\text{min}}}, \quad (13)$$

where Q_{curr} is the current retransmission count in the transmission queue, \bar{Q} is the average retransmission



count. Assume that the measured Q values on the path are (Q_1, Q_2, \dots, Q_m) , we have $\bar{Q} = \frac{1}{m} \times \sum_{n=1}^m Q_i$. Meanwhile, Q_{\max} and Q_{\min} can be calculated by

$$Q_{\max} = \underbrace{Q_1, Q_2, \dots, Q_m}_{\max}, Q_{\min} = \underbrace{Q_1, Q_2, \dots, Q_m}_{\min}. \quad (14)$$

Thus, we can simply use Q^Δ to probe whether or not the path d_{cp} to be with unreliable transmission condition (high loss). We determine d_{cp} to be with unreliable transmission condition if $Q^\Delta > 0$ and the value of Q^Δ is increasing continuously, namely,

$$\begin{cases} Q^\Delta > 0; \\ (Q^\Delta)_{t_n} - (Q^\Delta)_{t_{n-1}} > 0, \end{cases} \quad (15)$$

which t_n and t_{n-1} ($t_n > t_{n-1}$) are the observed time.

By cognizing the transmission condition of d_{cp} , once a SACK arrived at the sender, RAF-rtx attempts to trigger faster retransmissions for a missing data chunk following the rules below:

- (1) If $Q^\Delta > 0$ and $(Q^\Delta)_{t_n} - (Q^\Delta)_{t_{n-1}} > 0$, a DATA chunk with two consecutive miss indications (CMIs) will be marked for fast retransmission.
- (2) Otherwise, RAF-rtx refers to the regular SCTP operations where a DATA chunk with three CMIs will be marked for retransmission.

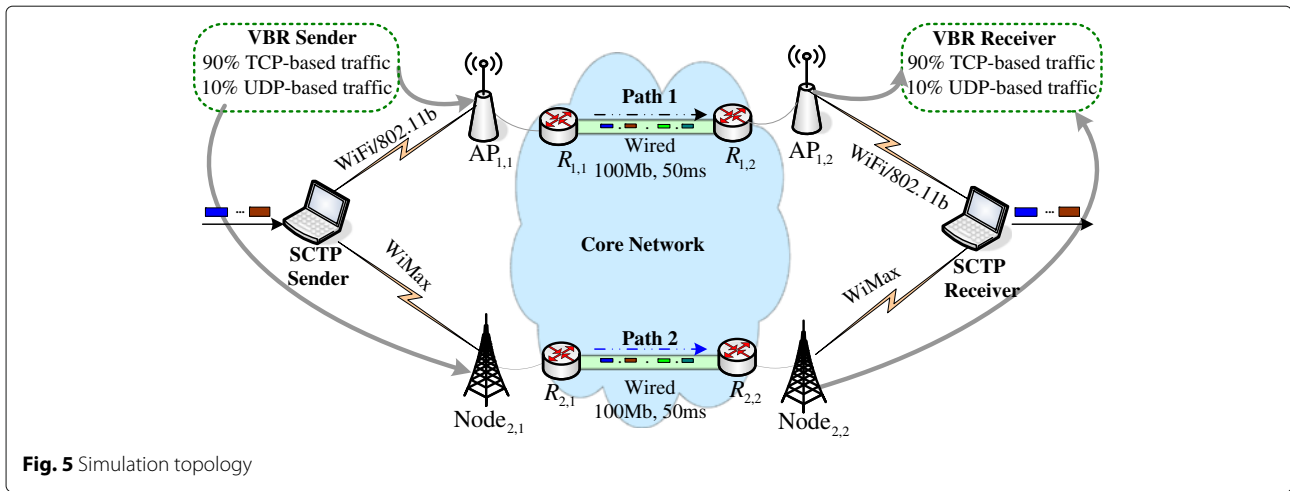
- (3) RAF-rtx does not mark the DATA chunk(s) with one miss indication for fast retransmission in order to avoid spurious retransmission.

3 Simulations and analysis

3.1 Simulation topology

The performance evaluation have been carried out using NS 2.35 (Network Simulator version 2.35) [34]. The simulations topology considered a SCTP-based HetNetEnv shown in Fig. 5. Both SCTP endpoints have two asymmetric paths (denoted path 1 and path 2) with different networking parameters. Path 1's bandwidth is set to 11 Mbps and 10–20 ms propagation delay, which is representative for a WiFi/IEEE 802.11b link. Path 2 experiences 10 Mbps bandwidth with 10–20 ms propagation delay, which is representative for a cellular WiMax/IEEE 802.16 link. Table 1 presents the major configuration of the two paths. The receive buffer (rbuf) size is set to a default value of 64 KB. The other SCTP default parameters just use the default values in NS 2.35. The simulation runs for 120 s of simulated time.

In order to simulate the wireless loss at data-link layer, we attach two loss models for each wireless access link, which are the *uniform loss model* that represents distributed loss caused by wireless interference and the *two-state Markov loss model*¹ that represents continuous loss caused by bursty traffic or signal fading. Moreover, we attach each path with a variable bit rate (VBR) generator in order to schedule VBR-based competing traffic over the two paths. Like [18], the packet sizes used for VBR traffic are selected as follows: 50 % are 44-byte, 25 % are 576-byte, and the rest 25 % have 1500-byte length, in which 20 % of these traffics are over UDP protocol and the other 80 % are carried by TCP protocol. Each path's VBR competing



traffic consumes randomly between 0 and 50 % of wireless WiFi/WiMax bandwidth.

3.2 Results analysis

1) Sending and receiving TSN: Fig. 6 shows the *sending and receiving TSN* (sr-TSN) when using SCTP, SCTP-Rev, and SCTP-Rev⁺, respectively. It can be observed that SCTP-Rev⁺ attains the highest number of sr-TSN, while SCTP attains the lowest number of sr-TSN. This is because the standard SCTP employs a simple failover-based primary path handover mechanism; in this way, it always utilizes the default primary path for data transport even if the transmission quality of the primary path is unsatisfactory. SCTP-Rev and SCTP-Rev⁺ achieve higher sr-TSN than SCTP because of their *receiver-based sending rate estimator* and *receiver-assisted path handover trigger*. However, benefiting from its *retransmission-aware fast retransmit module*, SCTP-Rev⁺ can detect the lossy condition and trigger a fast retransmission for a lost packet. Correspondingly, SCTP-Rev⁺ maintains the transmission efficiency at a higher level than the SCTP-Rev scheme.

2) Out-of-order TSN: The *out-of-order TSN* (o^3 -TSN), which is computed by the difference between the TSNs of two chunks received consecutively by the receiver,

is a good metric used to reflect the performance and characteristics of heterogeneous multipath transport scheme. Figure 7 illustrates the o^3 -TSN variations under the whole simulation time of 120 s. As shown in the figure, SCTP-Rev⁺ generates less o^3 -TSN and requires decreased reordering overhead than the SCTP and SCTP-Rev schemes. This is because that SCTP-Rev⁺ uses the receiver-based sending rate estimator and receiver-assisted path handover trigger to monitor per-path sending rate and adaptively select an optimal primary path for data delivery. Apart from the receiver-based sending rate estimator and receiver-assisted path handover trigger modules, SCTP-Rev⁺ also benefits from the retransmission-aware fast retransmit module which triggers a fast retransmission for loss recovery. The peak o^3 -TSN reception at the receiver is approximately 24, 17, and 14 when sending the data with SCTP, SCTP-Rev, and SCTP-Rev⁺, respectively.

3) Jitter Comparison: Jitter is a variation in packet transit delay caused by path handover, queuing, and other networking-related effects on transmission efficiency. Higher levels of jitter are more likely to occur on unsatisfactory transport solution. Therefore, jitter is an important metric to reflect the performance of multipath transport protocols. Figure 8 presents the jitter comparison results when using SCTP, SCTP-Rev, and SCTP-Rev⁺, respectively. The simulation results only between simulation time $t = 10$ s and $t = 20$ s are illustrated in the figure in order to better show the comparison. As the figure shows, all the three schemes experience delay jitter during data transmission. However, SCTP-Rev presents a lower packet transit delay variation than other two schemes, since it can detect a packet loss event timely and trigger a proper fast retransmission strategy for loss recovery. Moreover, SCTP-Rev is capable of acquiring and exploiting first-hand network parameters at receiver for primary path selection and handover operations. Thanks

Table 1 Per-path configuration used in the simulation

Parameters	Path 1	Path 2
Wireless access technology	IEEE 802.11 b	IEEE 802.16
The bandwidth of access link	11 Mbps	10 Mbps
The delay of access link	10–20 ms	10–20 ms
The queue length of access link	50 packets	50 packets
Uniform loss-rate	0–10 %	0–5 %
Markov loss rate	1 %	1 %
Core network bandwidth	100 Mbps	100 Mbps
Core network delay	50 ms	50 ms

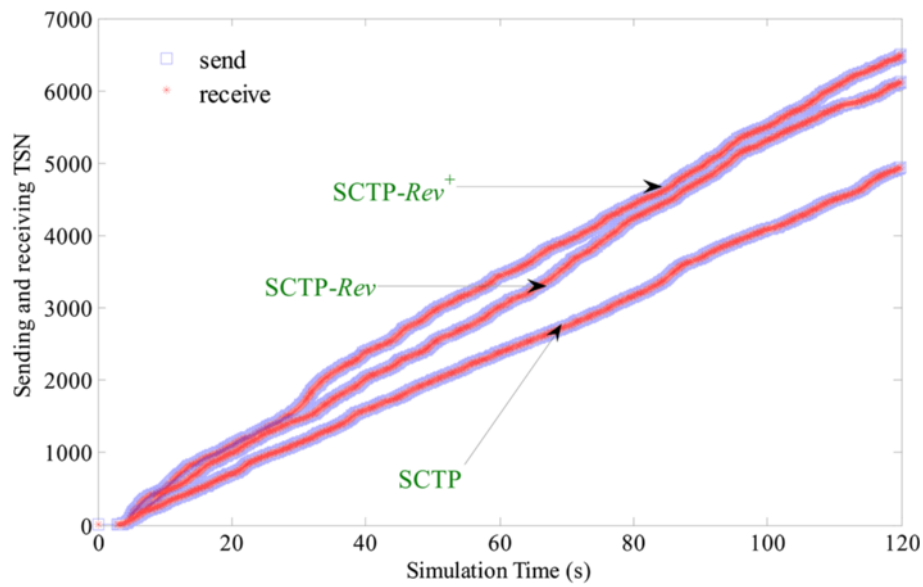


Fig. 6 The sr-TSN comparison

to these features, Sctp-Rev has an excellent potential for providing smooth and high-quality data delivery service.

4) *Throughput Comparison*: Figs. 9 and 10 present the average throughput performance comparison when Sctp, Sctp-Rev, and Sctp-Rev⁺ are used, when rbuf are 64, 32, and 128 KB, respectively. Figure 10b presents the cumulative average throughput of the three schemes with a total 120 s simulation time. As the figures show, Sctp-Rev⁺ achieves a higher throughput performance than the Sctp

and Sctp-Rev solutions in all the different rbuf sizes studied. That is because Sctp-Rev⁺ selects and switches to an optimal primary path, it also provides an adaptive fast recovery to handle packet loss. These factors help Sctp-Rev⁺ to increase the throughput performance in a varying HetNetEnv. After a total 120-s simulation time with a 64-KB rbuf, Sctp-Rev⁺'s throughput is 32.38 and 5.8 % higher than that of Sctp and Sctp-Rev, respectively. With a 32-KB rbuf is used, Sctp-Rev⁺ performs 37.13 and 10.27 % better than Sctp and Sctp-Rev,

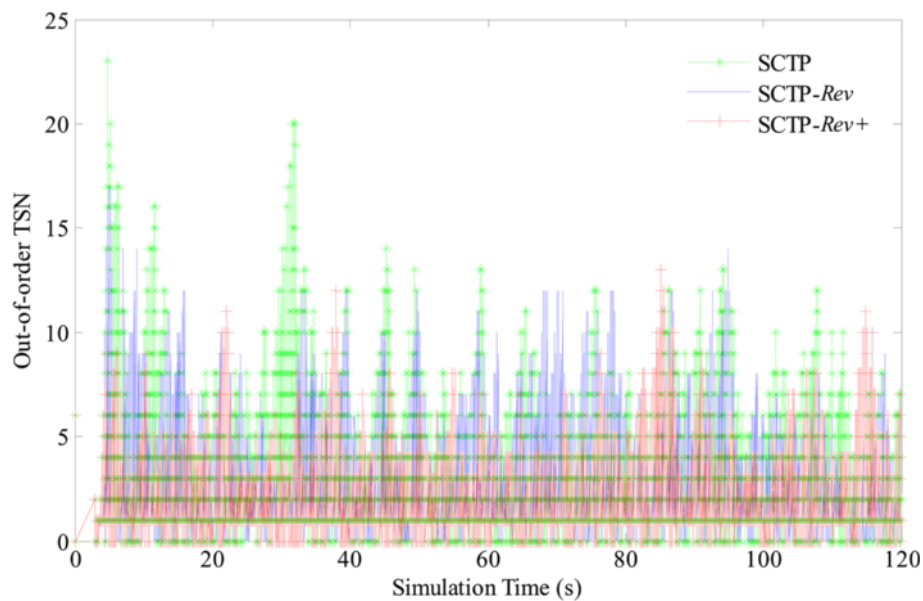


Fig. 7 The o³-TSN comparison

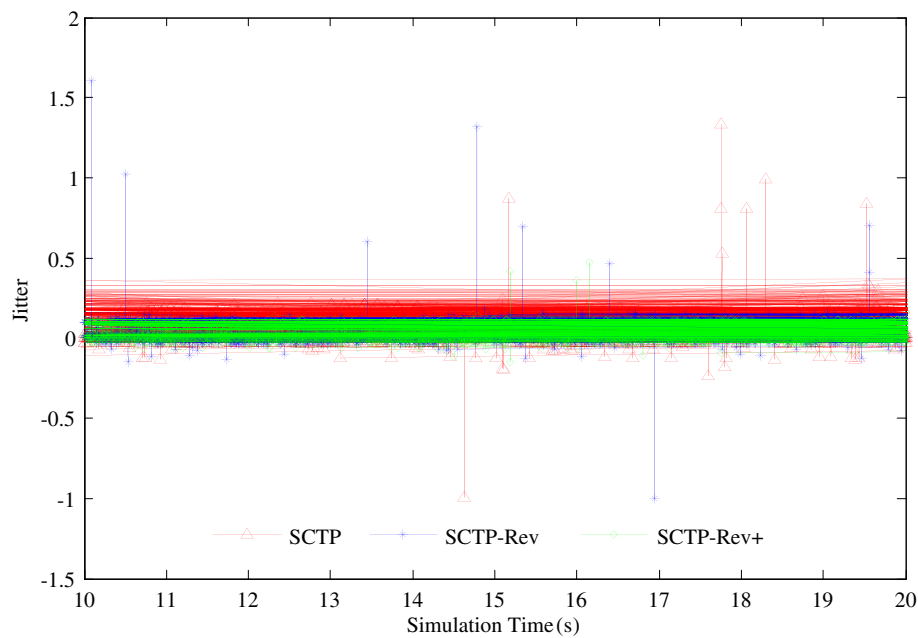


Fig. 8 Comparison of jitter

respectively. When the rbuf size is set to 128 KB, Sctp-Rev⁺ performs 38.74 and 5.63 % better than Sctp and Sctp-Rev, respectively.

As discussed previously, Sctp-Rev⁺ utilizes a WMA-based AIMD rate control mechanism to run flow control for each path and preserve fairness towards TCP-like network flows, by cutting the sending rate down to half on the

congested paths. In order to convince TCP friendliness of Sctp-Rev⁺, we examine the cumulative average TCP throughput performance when the default rbuf sizes of 64 KB is used, when Sctp, Sctp-Rev, and Sctp-Rev⁺ are used, respectively. After a total 120 s of simulation time, we find that the cumulative average throughput of Sctp-Rev⁺'s competing TCP traffic is only 0.17 % lower than

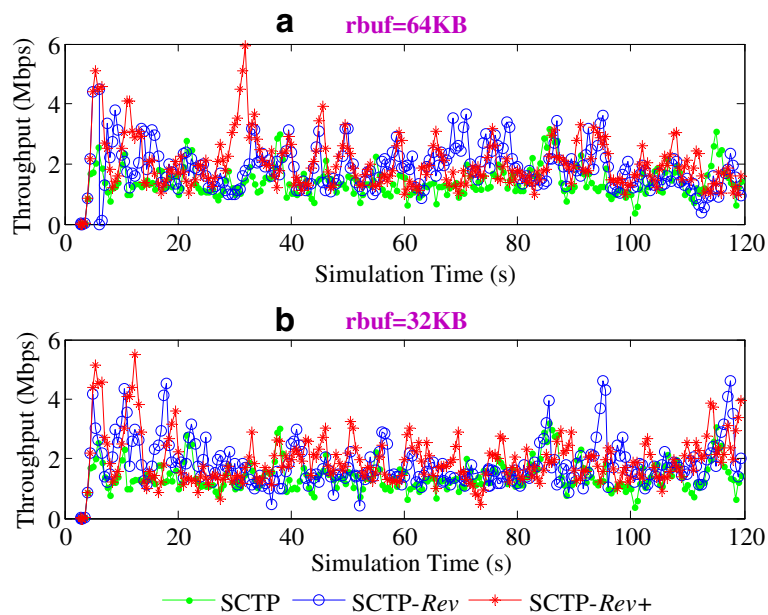


Fig. 9 Comparison of throughput with **a** rbuf = 64 KB and **b** rbuf = 32 KB

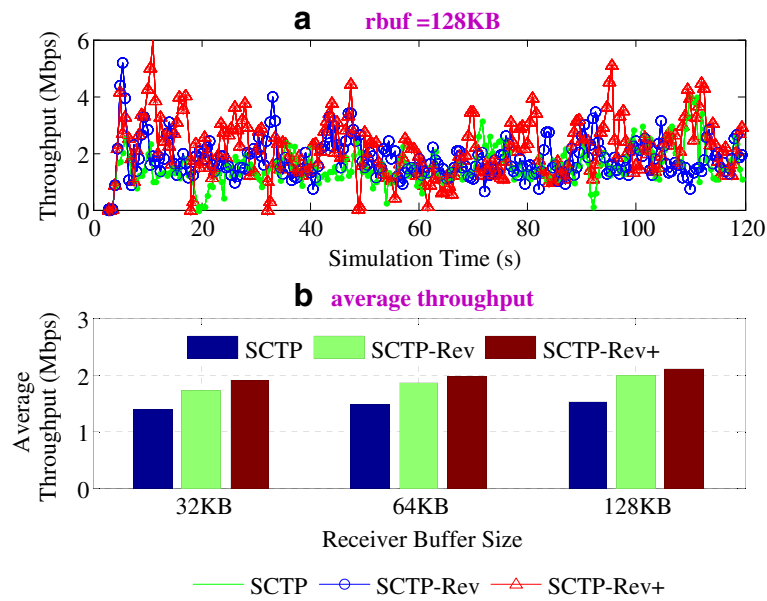


Fig. 10 a Throughput comparison (rbuf = 128 KB) and **b** average throughput

that of Sctp-Rev's competing TCP traffic and approximately 0.83 % lower than that of Sctp's competing TCP traffic. We argue that this TCP-aggressive behavior of Sctp-Rev⁺ is acceptable and unarmful. Therefore, we can conclude that Sctp-Rev⁺, also Sctp-Rev, can ensure TCP friendliness almost the same as Sctp does.

5) *Multimedia delivery comparison*: In order to better study the performance of Sctp-Rev⁺-based multimedia delivery, a well-known YUV video sequence (QCIF format, 176 × 144 pixels), which is composed of total 2000 frames with average streaming quality is used for the experimental video trace. Table 2 presents the encoding information on the first nine frames of the video trace file used in the simulation. The frame rate of the video stream is encoded at 30 frames per second (fps).

Table 2 The video encoding information used in the simulation

Frame number	Frame type	Frame size	Segment number	Time offset from the start of video (ms)
0	H	29	1 segment at	33
1	I	1508	2 segment at	66
2	P	254	1 segment at	99
3	B	223	1 segment at	132
4	B	204	1 segment at	165
5	P	343	1 segment at	198
6	B	290	1 segment at	231
7	B	336	1 segment at	264
8	I	1616	2 segment at	297

After processing, an MPEG-4 video sequence which consists of total 223 I frames, 445 P frames, and 1332 B frames is generated. These frames are included into 2250 packets (463 packets for I frame, 453 packets for P frame, and 1334 packets for B-frame) and are introduced to the NS2 by using an MPEG-4 video trace file. The simulation will stop when all the multimedia data are transmitted.

Figure 11 illustrates the comparison of multimedia delivery, expressed in terms of PSNR (dB) and received frame number, when Sctp, Sctp-Rev, and Sctp-Rev⁺ are used, respectively. Thanks to its highly efficient primary path selection/handover and loss-aware fast retransmission approaches, Sctp-Rev⁺ selects a path with a higher transmission efficiency as the primary path and schedules multimedia packets over the selected primary path; moreover, it can possibly avoid throughput performance degradation caused by fast retransmission/recovery using the current Sctp solutions. As a result, Sctp-Rev⁺ achieves a better multimedia delivery performance in comparison with Sctp and Sctp-Rev. The average PSNR difference between Sctp-Rev⁺ and Sctp is approximately 2.9 dB, and it is around 0.20 dB between Sctp-Rev⁺ and Sctp-Rev. The total received frame number of Sctp-Rev⁺ is 1832 (206I + 428P + 1220B), 1352 (165I + 305P + 880B) when using Sctp and 1756 (194I + 388P + 1172B) when using Sctp-Rev.

4 Conclusions

Motivated by the fact that a receiver may be adjacent to the wireless last hop and it is thus more

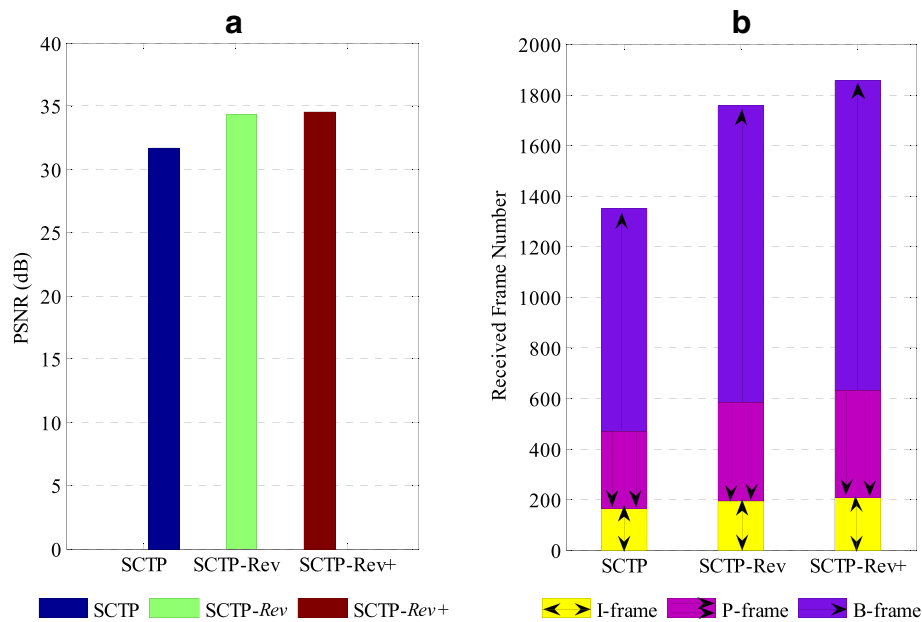


Fig. 11 Comparison of multimedia delivery **a** PSNR (dB) and **b** received frame number

aware of the wireless network parameters than its corresponding sender, this paper proposes *SCTP-Rev⁺*, a new receiver-assisted cellular/WiFi handover management mechanism for SCTP-based multimedia transport. *SCTP-Rev⁺* mainly consists of a *receiver-based sending rate estimator* (SRE-*rev*), a *receiver-assisted path handover trigger* (PST-*rev*), and a *retransmission-aware fast retransmit* (RAF-*rtx*) module, in which (i) SRE-*rev* is devoted to calculating the sending rate for each path at receiver side and giving the sender feedback on per-path sending rate; (ii) PST-*rev* runs primary path selection at receiver side and returns the selected primary path information to the sender for handover operations; and (iii) RAF-*rtx* is devoted to possibly detecting and handling packet loss timely.

Although simulation results reveal that applying *SCTP-Rev⁺* for multimedia delivery and handover is very useful, the energy consumption problem when applying *SCTP-Rev⁺* to the power-limited mobile devices is an interesting topic worth further investigation. We notice that recent research efforts [35] have been devoted to handling the energy consumption problem of multipath transport protocols by applying the promising cross-layer technology at the sender. Our future work will extend the *SCTP-Rev⁺* solution by running the cross-layer activities at receiver towards better reducing the energy consumption of *SCTP-Rev⁺*, and an open-source (Linux/Unix) virtual machine [36]-based performance analysis will be also included in our future work.

Endnotes

¹In the two-state Markov loss model, the packet loss process during data delivery is represented as a discrete-time random process with two-state Markov chain property. The interested reader, looking for the detail usage of Markov loss model used in a SCTP context, should refer to the reference [12] for more information.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61562044, 61262014, 71663037; the Natural Science Foundation of Jiangxi Province under Grant No. 20161BAB212046; the Project of Soft Science Research Plan of JiangXi Province under Grant No. 20161BBA10010; the Science and Technology Research Project of Jiangxi Provincial Department of Education (GJJ150319); and the Higher School Teaching Reform Research Subject of Jiangxi Province (JXJG-15-2-35).

Competing interests

The authors declare that they have no competing interests.

Author details

¹Jiangxi Normal University, 330022 Nanchang, China. ²Jiangxi Innovation Funds Management Center for Small and Medium-sized Enterprises, Nanchang, China.

Received: 31 March 2016 Accepted: 15 September 2016

Published online: 26 September 2016

References

1. J Wu, C Yuen, M Wang, J Chen, Content-aware concurrent multipath transfer for high-definition video streaming over heterogeneous wireless networks, *IEEE Trans. Parallel Distrib. Syst.* **27**(3), 710–723 (2016)
2. F Song, R Li, H Zhou, Feasibility and issues for establishing network-based carpooling scheme. *Pervasive Mob.Comput.* **24**, 4–15 (2015)

3. F Song, H Zhou, S Zhang, H Zhang, I You, The throughput critical condition study for reliable multipath transport. *Comput.Sci. Inf. Syst.* **10**(2), 567–787 (2013)
4. Y Cao, C Xu, J Guan, H Zhang, CMT-CC: Cross-layer cognitive CMT for efficient multimedia distribution over multi-homed wireless networks. *Wirel.Pers. Commun.* **82**(3), 1643–1663 (2015)
5. R Stewart, Stream control transmission protocol. IETF RFC 4960 (Proposed Standard), Sep. 2007
6. J Baek, PS Fisher, M Jo, H-H Chen, A lightweight SCTP for partially reliable overlay video multicast service for mobile terminals. *IEEE Trans.Multimedia.* **12**(7), 754–766 (2010)
7. Y Cao, C Xu, J Guan, H Zhang, Qos-driven SCTP-based multimedia delivery over heterogeneous wireless networks. *Sci.China Inf. Sci.* **57**(10), 1–10 (2014)
8. J Wu, B Cheng, C Yuen, et al., Distortion-aware concurrent multipath transfer for mobile video streaming in heterogeneous wireless networks. *IEEE Trans. Mob. Comput.* **14**(4), 688–701 (2015)
9. C Xu, Z Li, J Li, H Zhang, G Muntean, Cross-layer fairness-driven concurrent multipath video delivery over heterogenous wireless networks. *IEEE Trans.Circ. Syst. Video Technol.* **25**(7), 1175–1189 (2015)
10. J Wu, C Yuen, N Cheung, J Chen, Delay-constrained high definition video transmission in heterogeneous wireless networks with multi-homed terminals. *IEEE Trans.Mob. Comput.* **15**(3), 641–655 (2016)
11. Y Kim, S Lee, MSCTP-based handover scheme for vehicular networks. *IEEE Commun. Lett.* **15**(8), 828–830 (2011)
12. TD Wallace, A Shami, Concurrent multipath transfer using SCTP: modelling and congestion window management. *IEEE Trans. Mob. Comput.* **13**(11), 2510–2523 (2014)
13. C Xu, T Liu, J Guan, H Zhang, G-M Muntean, CMT-QA: quality-aware adaptive concurrent multipath data transfer in heterogeneous wireless networks. *IEEE Trans. Mob. Comput.* **12**(11), 2193–2205 (2013)
14. J Wu, Y Shang, J Huang, X Zhang, B Cheng, J Chen, Joint source-channel coding and optimization for mobile video streaming in heterogeneous wireless networks. *EURASIP J.Wirel. Commun. Netw.* **2013**, 283 (2013)
15. V Anil Kumar, D Das, Data Enriched SACK: a novel acknowledgement generation scheme for secure SCTP. *IEEE Commun. Lett.* **18**(12), 2109–2112 (2014)
16. F Song, M Xue, S Zhang, Security model for analyzing data privacy in multipath transport. *China Commun.* **9**(5) (2012)
17. Y Wang, R Injong, H Sangtae, in *Proc. of IEEE INFOCOM Workshops*. Augment SCTP multi-streaming with pluggable scheduling (IEEE, Piscataway, 2011)
18. G Sarwar, P-U Tournoux, R Boreli, E Lochin, in *Proc.of IEEE LCN*. eCMT-SCTP: improving performance of multipath SCTP with erasure coding over lossy links (IEEE, Piscataway, 2013)
19. M Becke, H Adhari, EP Rathgeb, et al., in *Proc.of IEEE GLOBECOM*. Comparison of multipath TCP and CMT-SCTP based on intercontinental measurements, (2013), pp. 1360–1366
20. J Wu, Y Chau, B Cheng, Y Yang, M Wang, J Chen, Bandwidth-efficient multipath transport protocol for quality-guaranteed real-time video over heterogeneous wireless networks. *IEEE Trans. Commun.* **64**(6), 2477–2493 (2016)
21. L Zhou, Z Yang, Y Wen, et al., Distributed wireless video scheduling with delayed control information. *IEEE Trans.Circ. Syst. Video Technol.* **24**(5), 889–901 (2014)
22. L Zhou, X Wang, W Tu, et al., Distributed scheduling scheme for video streaming over multi-channel multi-radio multi-hop wireless networks. *IEEE J.Sel. Areas Commun.* **28**(3), 409–419 (2010)
23. S McCanne, V Jacobson, M Vetterli, in *Proc. of ACM SIGCOMM*. Receiver-driven layered multicast (ACM, New York, 1996), pp. 117–130
24. I Rhee, V Ozdemir, Y Yi, TEAR: TCP Emulation at receivers-flow control for multimedia streaming. Technical Report, North Carolina State University
25. L Budzisz, J Garcia, A Brunstrom, et al., A taxonomy and survey of SCTP research. *ACM Comput. Surv.* **44**(4), 1–36 (2012)
26. Y Cao, C Xu, J Guan, H Zhang, in *Proc. of IEEE International Conference on Multimedia & Expo*. Receiver-driven SCTP-based multimedia streaming services in heterogeneous wireless networks, (ICME, 2014) (IEEE, Piscataway, pp. 1–6
27. D Chiu, R Jain, Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. *Comput. Netw. ISDN Syst.* **17**(1), 1–14 (1989)
28. C-M Huang, C-H Tsai, The handover control mechanism for multi-path transmission using Stream Control Transmission Protocol (SCTP). *Comput. Commun.* **30**(17), 3239–3256 (2007)
29. S Mascolo, C Casetti, et al., TCP Westwood: bandwidth estimation for enhanced transport over wireless links. *Proc.of ACM SIGMOBILE* (2001)
30. J Liao, J Wang, X Zhu, cmpSCTP: an extension of SCTP to support concurrent multi-path transfer. *Proc.of IEEE ICC*, 5762–5766 (2008)
31. F Song, Y Zhang, Z An, H Zhou, I You, The correlation study for parameters in four tuples. *Int.J. Ad Hoc Ubiquit. Comput.* **19**(1/2), 38–49 (2015)
32. K Darabkh, I Jafar, G Sukkar, et al., An improved queuing model for packet retransmission policy and variable latency decoders. *IET Commun.* **6**(18), 3315–3328 (2012)
33. X Cheng, P Mohapatra, in *Proc.of IEEE WCNC*. Retransmission-aware queuing and routing for video streaming in wireless mesh networks (IEEE, Piscataway, 2011), pp. 2137–2142
34. UC Berkeley, LBL, USC/ISI and Xerox Parc, NS-2 documentation and software (2011). www.isi.edu/nsnam/ns/
35. J Wu, C Yuen, B Cheng, M Wang, J Chen, Energy-minimized multipath video transport to mobile devices in heterogeneous wireless network. *IEEE J. Sel. Areas Commun.* **34**(5), 1160–1178. (2016.)
36. F Song, D Huang, H Zhou, H Zhang, I You, An optimization-based scheme for efficient virtual machine placement. *Int.J. Parallel Prog.* **42**(5), 853–872 (2014)

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com