



HAL
open science

eCMT-SCTP: Improving Performance of Multipath Sctp with Erasure Coding Over Lossy Links

Golam Sarwar, Pierre-Ugo Tournoux, Roksana Boreli, Emmanuel Lochin

► To cite this version:

Golam Sarwar, Pierre-Ugo Tournoux, Roksana Boreli, Emmanuel Lochin. eCMT-SCTP: Improving Performance of Multipath Sctp with Erasure Coding Over Lossy Links. IEEE 38th Conference on Local Computer Networks (LCN 2013), Oct 2013, Sydney, Australia. pp.0. hal-04041213

HAL Id: hal-04041213

<https://hal.science/hal-04041213>

Submitted on 22 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Open Archive Toulouse Archive Ouverte (OATAO)

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <http://oatao.univ-toulouse.fr/>
Eprints ID: 9593

To cite this document: Sarwar, Golam and Tournoux, Pierre-Ugo and Boreli, Roksana and Lochin, Emmanuel *eCMT-SCTP: Improving Performance of Multipath SCTP with Erasure Coding Over Lossy Links*. (2013) In: IEEE 38th Conference on Local Computer Networks (LCN 2013), 21 October 2013 - 24 October 2013 (Sydney, Australia).

Any correspondence concerning this service should be sent to the repository administrator: staff-oatao@inp-toulouse.fr

eCMT-SCTP: Improving Performance of Multipath SCTP with Erasure Coding Over Lossy Links

Golam Sarwar^{†,*}, Pierre-Ugo Tournoux^{◇,▷}

^{*} National ICT Australia Ltd, ^{*} University of NSW, Australia.

[◇] IREMA, LIM, Université de la Réunion, France

[†]golam.sarwar@nicta.com.au, [▷]pierre.tournoux@univ-reunion.fr

Roksana Boreli^{‡,*}, Emmanuel Lochin^{◇,◁}

[◇]Université de Toulouse,

ISAE, LAAS-CNRS, France.

[‡]roksana.boreli@nicta.com.au [◁]emmanuel.lochin@isae.fr,

Abstract—Performance of transport protocols on lossy links is a well researched topic, however there are only a few proposals which make use of the opportunities presented by the use of erasure coding within the transport layer, particularly in the multipath transport protocol context. In this paper, we investigate the improvements in the performance of multipath SCTP transport protocol brought by the novel integration of the on-the-fly erasure code into the congestion control and reliability mechanisms of CMT-SCTP. Our contributions include: integration of transport protocol and erasure codes with regards to congestion control and reliability mechanisms; proposal for a variable retransmission delay parameter in the sender (*aRTX*) adjustment; evaluation of the performance of CMT-SCTP with erasure coding with simulations. Our results show that we can achieve from 10% to 80% improvements in application goodput under lossy multipath network conditions without a significant penalty i.e. with a minimal (10%) overhead due to the encoding-decoding process. We further evaluate the performance of video streaming using an equivalent of partially reliable CMT-SCTP with erasure coding and again demonstrate a solid performance improvement for our proposal, compared to CMT-SCTP. Finally, we provide an analytical evaluation of CMT-SCTP with erasure codes and validate this with simulation results.

Index Terms—Multipath Transport Protocol; Erasure Coding; CMT-SCTP;

I. INTRODUCTION AND MOTIVATION

Most modern mobile devices are equipped with multiple network interfaces, with a common combination of 3G and Wi-Fi. This ensures the ubiquitous availability of a network connection for the users and additionally enables them to manage costs by using e.g. free hotspot or in-home Wi-Fi connectivity. The widespread of heterogeneous connectivity necessitates changes in the well-established protocols, to fully exploit this diverse networking environment. Multipath transport protocols aim to provide the next generation data transmission over these readily available (and most commonly wireless) network paths, to efficiently exploit multi-homing and multipath based parallel data transfer. In line with this, IETF has been progressing the work towards a multipath networking capable version of TCP, MPTCP [1]; similarly for SCTP [3] (designed to overcome the shortcomings of traditional transport protocols like TCP), a Concurrent Multipath Transfer (CMT-SCTP) extension [2] has been proposed.

Reliable transport protocols such as TCP and SCTP perform poorly in presence of lost data packets [4], [5]. Both TCP and SCTP react to any lost data packet by reducing the

sender's data emission rate and by using fast recovery and time-out based retransmissions [3]. Transport layer's inability to distinguish between packet loss due to congestion in the network and lower layer data loss makes it even more difficult for the transport layer to perform well on lossy links. A potential improvement may be achieved by the use of explicit congestion notification (ECN) [6] provided by the intermediate routers, by which the congestion is explicitly indicated by including marks in the packet header. However, ECN does not solve the transport protocol issues in regards to link losses, which can cause significant performance penalty for the applications requiring sufficient throughput and in-time data delivery. In this paper, we focus on improving the performance of data transmission over lossy links, with application to wireless multipath data transfer. Changing conditions in mobility scenarios for mobile devices with heterogeneous wireless connectivity, including both short connectivity losses on all links due to hand-offs and erroneous wireless links, resulting in a varying magnitude of losses that could significantly impair the performance of applications, create likely scenarios for demonstrating the improvements achievable by our proposal. Although we primarily focus on the multipath version of the SCTP protocol, CMT-SCTP, [2], our proposal is equally applicable to other, both single-path and multipath transport protocols.

Compared to the existing work on improving performance of multipath TCP and SCTP with various error correcting codes in a lossy environment, our key contributions presented in this paper are as follows.

We study the feasibility and the potential benefits of CMT-SCTP with erasure codes. We consider three different types of erasure codes, i.e. block codes, convolutional codes and on-the-fly erasure codes integrated within CMT-SCTP and we evaluate the performance of such a system for generic web applications using fully reliable CMT-SCTP and for video streaming using an equivalent of partially reliable CMT-SCTP.

To further improve the performance, we propose a modification of the retransmission mechanism at the SCTP data sender with a variable retransmission delay (*aRTX*), based on the type of error correction code. This includes a method to estimate the variable sender retransmission delay adjustment (*aRTX*). We finally present an analysis of CMT-SCTP performance with erasure codes based on the model from [21],

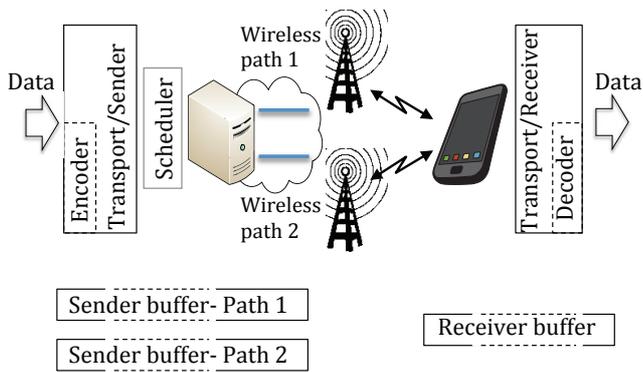


Figure 1. Architecture of the integrated multipath transport and erasure coding

validated against simulation results for a range of packet loss conditions.

The organization of the rest of the paper is as follows: in Section II, we elaborate our proposed integration of erasure codes with the transport layer and the adaptive retransmission scheme (*aRTX*). In Section III, we present an evaluation of our proposal, and demonstrate improvement both in terms for goodput for generic web applications and in terms of quality for video streaming. We provide an evaluation of eCMT-SCTP's performance validated against a multi-homed SCTP throughput model in Section IV. In Section V, we discuss the prior work and we present conclusions in Section VI.

II. OUR PROPOSAL

The changing conditions in the wireless environment, including short connectivity loss in mobility situations, result in a decreased performance of transport protocols and, consequently, in a lower quality of experience for the users. Erasure codes have been designed to handle lossy conditions and have been used in other proposals conjointly with transport protocols, e.g. in [8]. Our goal is to consider the potential benefits of this approach in situations where multiple wireless paths can be utilized for common mobile applications like web browsing and video streaming. To improve the multipath transport performance in conditions of varying network loss, we propose eCMT-SCTP, which integrates erasure codes within the CMT-SCTP transport protocol. To further improve the performance of this integration, we propose a sender side modification to the packet retransmission scheme (*aRTX*).

In the CMT extension of SCTP [2], a single logical data connection can simultaneously use multiple physical network paths. In line with SCTP mechanism, congestion control uses a TCP-like window based algorithm, however a separate sender congestion control window (CWND) is allocated to each path. A single receiver window (RWND) is used for all paths. As per SCTP, CMT-SCTP includes partially and fully reliable transport with in-order or out of order packet delivery. For reliable transport with in-order delivery, which we focus on in this work, receiver uses selective acknowledgement (SACK) packets to acknowledge successful reception of data packets.

The proposed architecture to integrate erasure coding within the transport layer is shown in Figure 1. The central idea is to introduce redundancy inside the transmitted data flow at transport layer, applicable to all physical paths. Here, we show a multi-homed mobile with two wireless interfaces, connected to a server using two simultaneous network paths.

We assume that the imbalance between available paths (i.e. the difference in capacity and/or end-to-end delay) is mitigated either by intelligent scheduling at the sender side, as proposed in [27] or by a split buffer based solution proposed in [28]. In Figure 1, we show the advanced scheduler module, that allocates data to the different paths according to both the available window size and the difference in end-to-end delay on individual paths. Therefore, when evaluating the performance, we focus solely on evaluating the impact of lossy links.

In the proposed scheme, based on the specified encoding parameters, the sender encodes data packets and produces redundant packets as linear combinations of the data packets from the sender's buffer(s). The encoding process ensures that the data packets for which a redundant packet is produced, have not already been acknowledged by the receiver, i.e. that redundant packets are generated only for the packets which are in the flight or were lost. A decoding at the receiver is attempted on arrival of every new packet. Once a successful decoding takes place, leading to recovery of a missing packet, a SACK packet is sent immediately to notify the sender. This is in contrast to the CMT-SCTP delayed SACK modification [2]. The (non-delayed) SACK allows the sender to immediately release buffered packets kept for retransmission leading to the increase of congestion window. For the case where the lost packet could not be recovered, the standard SCTP notification (gap report) is sent back to the sender.

We consider the use of on-the-fly convolutional codes and standard convolutional and block codes. The coding techniques are defined by the following parameters [12]. k is the number of input data packets to be encoded, n the number of output packets after encoding is performed and m defines the memory length for the convolutional and on-the-fly erasure codes.

Both block and convolutional codes used in this paper are systematic. A block code is defined by (n, k) . For an (n, k, m) convolutional code, redundancy packet R is created from previous m input data packets from the sender buffer(s). On-the-fly erasure codes are a variation of convolutional codes, and TETRYS systematic erasure code [13] used in this paper can also be represented by (n, k, m) . Note the code's memory size m is variable and we can encode all available packets from the sender buffer into the encoded packet, as compared to the standard convolutional codes which have a fixed memory length.

A. Congestion Control, Reliability and Erasure Codes

In our proposal, redundancy packets are not considered by the congestion control and reliability mechanism of the transport protocol. Therefore, they do not occupy space in the sender's buffer, neither they are retransmitted if lost. It

should also be noted that we do not introduce any additional decoding buffer in our proposed scheme. Rather, the same receiver’s buffer is shared both by the data packets and the redundant packets. If the buffer is full, incoming data packets are prioritized over redundant packets. Similarly, if new redundant packets arrive, we perform a comparison between the redundant packet and the existing redundant packets in the buffer to quantify their utility given the current sequence of data packets in the buffer. The receiver-side buffer management algorithm when the buffer is full is presented in Algorithm 1.

Algorithm 1 Receiver Buffer Management

```

function FIND_LEAST_SIGNIFICANT
  LeastSig ← Null
  CumAck ← HighestAcknowledgedSequence
  for each Packet ← PacketsInRecvBuf do
    if Packet.Type = Data
      and NeededForDecoding(Packet) = False then
        LeastSig ← Packet
      end if
    if Packet.Type = Redundancy
      and EncodesPacketsBelow(CumAck) = True then
        LeastSig ← Packet
      end if
    end for
  if LeastSig = Null then
    LeastSig ← FindOldestRedundant(RecvBuf)
  end if
  if LeastSig = Null then
    Data ← FindOlderData(RecvBuf)
    if Data.Seq < CumAck then
      LeastSig ← Data
    end if
  end if
  if LeastSig = Null then
    return LeastSig
  end function
  //Packet Reception Loop at Data Receiver
  while Packet ← NewIncomingPackets do
    if RecvBuf is Full then
      LS ← Find_Least_Significant(Packet)
      if LS ≠ Null then
        RecvBuf[LS] ← Packet
      else
        Drop(Packet)
      end if
    else
      RecvBuf[Unoccupied] ← Packet
    end if
  end while

```

SCTP (and consequently CMT-SCTP) provides fully reliable, non-reliable [26] and partially reliable [22] transport options. Fully reliable SCTP provides reliability to an ongoing data flow by means of fast retransmissions and time-out retransmissions. If a packet is reported missing 4 times by the data receiver, SCTP sender will first attempt a fast recovery, while halving the congestion window. If the packet is still reported as missing, eventually the retransmission timer (RTO) will expire and a time-out retransmission will take place, consequently forcing SCTP into a slow-start phase for

the corresponding path. During the initial evaluation of the integrated erasure codes, we have observed that while many of the lost packets were actually recovered by the erasure code, the same packets were also spuriously retransmitted by the sender. This was due to the independent operation of the erasure recovery (decoder) and the transport protocol’s own retransmission based recovery mechanisms. To mitigate this, we propose an adaptive retransmission scheme which modifies the default fast and time-out mechanism of reliable SCTP with erasure codes.

B. Adaptive RTX

For the integrated erasure coding to be effectively utilised in reliable eCMT-SCTP, there has to be a sufficient delay in the SCTP retransmission mechanism to enable the lost packets to be recovered by the decoder, *i.e.* the system has to allow for the transmitting and receiving of the redundant packets which relate to the missing data packets. Therefore, our proposed adaptive retransmission (*aRTX*) scheme follows the below steps:

- 1) After a packet has been marked for fast retransmission, the retransmission is delayed at the sender by a timer value (in the number of packets) set to δ , which ensures that the missing packet is not retransmitted at least until the transmission of the next redundant packet, which encodes the missing packet and until one more gap-report for the same missing packet is received.
- 2) Once a time-out is triggered, before performing a time-out recovery, the sender checks if an existing fast recovery is pending due to *aRTX*. If yes, the sender performs a fast recovery recalculating RTO for the fast retransmitted packets. Otherwise, standard time-out retransmission is performed.

The sender-side algorithm for the above steps is presented in Algorithm 2. As will be demonstrated by our simulation results, these rules allow room for the erasure code to perform packet recovery without the penalty in congestion window or flow control improving the overall performance further.

C. Method Used to Estimate δ

For the case of reliable CMT-SCTP with erasure codes, the probability of spurious retransmissions can be approximated by the probability that the receiver will decode a lost packet after the receiver has transmitted a number of gap reports referring to the lost packet and the sender has retransmitted this packet. In *aRTX*, a sender will wait for n packets to be transmitted including the redundant packet before triggering retransmission, therefore we have:

$$\delta \approx \left(n \times L \times \frac{1}{C} \right) + \frac{3RTT}{2} \quad (1)$$

where n is the number of sender transmitted packets after lost packet was transmitted, L is the average packet size and C is the link capacity. This estimation holds in the case of block code (because of the fixed size of a block)

Algorithm 2 Adaptive RTX (*aRTX*)

```
//Packet Transmission Loop at Data Sender
while Packet ← NewOutgoingPackets do
  if Packet.Type = DATA then
    Packet.δ ← Estimate_Delta(Packet)
  end if
end while
//Packet Reception Loop at Data Sender
while Packet ← NewIncomingPackets do
  if Packet.Type = SACK then
    for each Missing in ReportedMissingPkts do
      if FastRtx is Due then
        if Missing.δ ≠ 0 then
          Missing.δ ← (Missing.δ -  $\frac{s}{C}$ )
          Continue //aRTX Delay
        else
          Fast_Recovery(Missing)
        end if
      end if
      if TimeoutRtx is Due then
        if Missing.δ ≠ 0 then
          Fast_Recovery(Missing)
        else
          Rto_Expiry(Missing)
        end if
      end if
    end for
  end if
end while
```

but not for convolutional and on-the-fly codes. In this case, our first attempts have shown that this estimation is complex to derive analytically. However, in the case of convolutional and on-the-fly codes, deriving an analytical estimation is not mandatory. Indeed, due to the SCTP sender congestion window progression, we know that it takes at least one RTT to transmit the whole window and at least one more RTT to receive 4 gap reports which trigger retransmission. Thus we propose to set δ to an upper bound corresponding to twice the RTT of the slowest path:

$$\delta = \min(2 * RTT_i) \text{ with } i \in 0, 1, \dots, n \quad (2)$$

where RTT_i corresponds to the RTT of path i . As a first step, we propose to use this easy to implement upper bound. In future work we plan to refine this value following the model proposed in [14].

III. EVALUATION OF ECMT-SCTP OVER LOSSY LINKS

We have implemented our eCMT-SCTP proposal under existing CMT-SCTP contribution in NS-2 [20]. We have also implemented ECN for CMT-SCTP under NS-2 as per the IETF draft [6]. The network topology for our experiments follows the configuration shown in Figure 1, with a multi-homed mobile that has two wireless interfaces and is downloading a single data stream over two simultaneous network paths. Following the assumptions about path asymmetry being handled by one of the existing proposals [27], [28], we use the following parameters for all experiments: bandwidth and RTT for each of the paths are, respectively, $1Mbit/s$ and $100ms$. Both

Path1 and *Path2* have a uniform packet loss, which is varied from 1% – 10%. Future work will consider fading channels with bursty losses, where on-the-fly coding scheme is known to perform well regardless of the configuration used [13]. All erasure codes used have equal redundancy of 10%, we used a (10, 9) block code and (10, 9, 20) convolutional code; with $m = 20$ being the initial value for on-the-fly convolutional code and the default size of the SCTP sender buffer, 43 packets, being the maximum m limit (as m varies for these codes). Therefore, out of $2Mbit/s$ aggregated available bandwidth over the two lossy paths, erasure coded CMT-SCTP had $1800Kbit/s$ effectively available, while the $200Kbit/s$ was dedicated to encoded redundant packets. Each of the experiments presented below was run 50 times with a random seed and each for a duration of $300sec$.

A. Fully Reliable eCMT-SCTP With Generic Web Traffic

We first consider the performance of reliable transport, with generic web traffic *i.e.* web browsing or file download. Simulation results for the application goodput and the percentage of duplicate packets are shown in Figures 2 and 3. All graphs include the values of mean and standard deviation.

1) *Single Stream Over Multipath without ECN*: We first present simulation results for a single-flow single-stream data transfer over two paths without support for ECN. Single-flow single-stream represents a multipath transfer where the overall data flow has a single sequence-number space. This is a challenging scenario for all current multipath protocols, as packets arriving out of order occupy receiver’s buffer eventually becoming a bottleneck blocking further data transmission. As can be seen in Figure 2, at around 2% packet loss we start achieving benefits of our erasure coding and *aRTX* proposals.

We can observe that the highest performance improvement is achieved with TETRYS and *aRTX*, followed by convolutional and block codes (also with *aRTX*). Although the absence of *aRTX* reduces the achievable performance gains, again TETRYS codes outperform convolutional and block codes in the simulated scenario. We note that, as shown in our previous work [9], CMT-SCTP fails to achieve the total aggregated throughput of $2Mbit/s$ due to SCTP’s recommended default delayed SACK algorithm and receiver’s buffer blocking even in the scenario which has no losses. This also applies to our eCMT-SCTP version, therefore restricting the total achievable goodput to around $1.5Mbit/s$.

2) *Single Stream Over Multipath with ECN*: We present simulation results of the same single-flow single-stream data transmission scenario, but over two ECN capable paths. As can be seen in Figure 3, although the ECN capable standard CMT-SCTP protocol demonstrates nearly the same pattern of degradation over lossy paths as shown in the non-ECN experiments, our proposed eCMT-SCTP outperforms the standard CMT-SCTP by a significant margin. It should be noted that in both of the scenarios TETRYS outperforms other erasure codes once the combined improvement ratio for goodput and reduction in spurious retransmission is considered. As shown in [13], TETRYS is actually expected to perform even better if

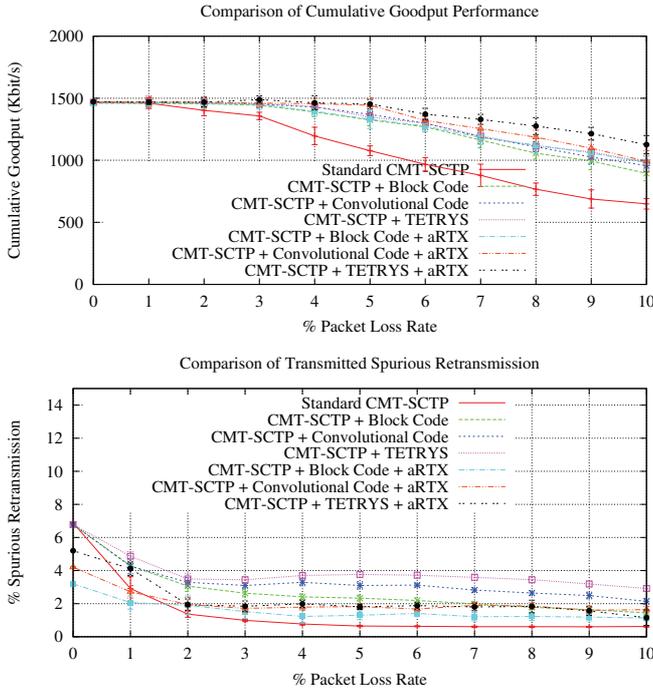


Figure 2. Performance comparison of standard CMT-SCTP and CMT-SCTP with erasure code and adaptive retransmission on ECN incapable network

the transport protocol itself does not introduce any bottleneck. As previously noted, single-flow single-stream data transfer over multiple paths is a challenging scenario for all currently proposed multipath protocols, as any imbalance in individual paths leads to out of order packet arrival and eventual buffer blocking [28], [9]. Therefore, we have purposefully focused this study on the improvements erasure codes could bring to a scenario in which paths are symmetrical, as the main aim (and utility) of the proposed scheme is in improving the performance of erroneous lossy paths. Enhancement to multipath protocols to overcome the path imbalance issue is one of the challenges we plan to study in future work.

B. Equivalent Partially Reliable eCMT-SCTP With Video Streaming

We now present the results for the performance of video streaming over the same simulation scenario from Figure 1. As CMT-SCTP (or SCTP) in NS-2 does not include the implementation of partial reliability as defined in [22], we use the fully reliable data transmission during simulations and introduce a delay constraint *i.e.* each packet that is delayed by more than d_{max} is discarded by the streaming video playback application. Although this evaluation does include unnecessary retransmissions, we consider it a good first order approximation of the performance of partially reliable eCMT-SCTP. We choose $d_{max} = 150ms$ as defined by ITU-T G.114 [23] and the same way used in [24]. Similarly to the evaluation of fully reliable eCMT-SCTP, each experiment was run 50 times, with a random seed for the simulated loss, and with a 300sec duration of each run. We use the processed results

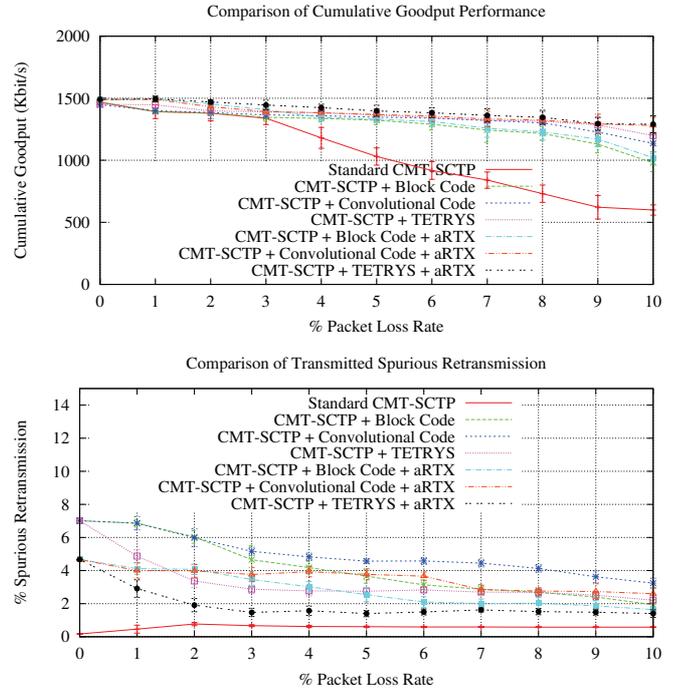


Figure 3. Performance comparison of standard CMT-SCTP and CMT-SCTP with erasure code and adaptive retransmission on ECN capable network

of NS-2 simulations (with delayed packets filtered with d_{max} as appropriate) as input to ITU-T G.1070 [25] recommended opinion model for evaluating video and telephony applications. G.1070 opinion model provides us with opinion scores as represented by V_q in Equation 3, where P_{plv} is the packet loss rate and D_{Pplv} is the degree of video quality robustness due packet loss in the application layer. Basic video quality I_c is calculated as shown in Equation 4. Video quality V_q with no packet loss in the network is expressed by Equation 5. b in these equations represents bit-rate of the encoded video and coefficients v_3 , v_4 and v_5 depends on the type of video codec used, video playback display format, intervals in video key-frames and video playback display size as per ITU-T recommendations in G.1070.

$$V_q = 1 + I_c \cdot e^{\frac{P_{plv}}{D_{Pplv}}} \quad (3)$$

$$I_c = v_3 \cdot \left(1 - \frac{1}{1 + \left(\frac{b}{v_4}\right)^{v_5}} \right) \quad (4)$$

$$V_q = 1 + v_3 \cdot \left(1 - \frac{1}{1 + \left(\frac{b}{v_4}\right)^{v_5}} \right) \quad (5)$$

In Figure 4, first we show the packet loss rate as experienced by the video playback application with respect to the late incoming packets with $d_{max} = 150ms$. Then we use these values to calculate the video quality evaluation opinion metric defined by the ITU-T G.1070 model [25]. Figure 5 shows

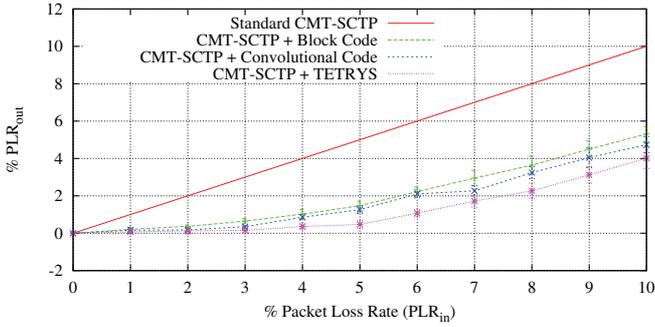


Figure 4. Packet error rate (PLR_{out}) as experienced by the video playback application with maximum tolerable delay for the incoming packets of $d_{max} = 150ms$

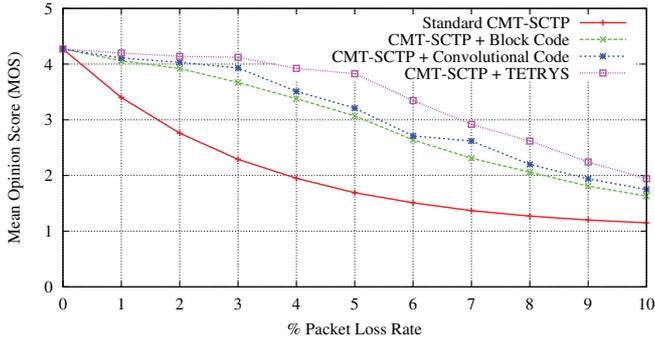


Figure 5. Mean Opinion Score (MOS) computed by ITU-T G.1070 model

the perceived video quality represented by Mean Opinion Score (MOS) for a range of packet loss ratios on the network link. For video quality evaluation, we have used standard MPEG4 video codec with QVGA resolution (Quarter VGA, 320×240 pixels), bit-rate $b = 1000kbit/s$, frame-rate $30fps$, key-frame interval of $1fps$ and with a video playback display size of $4.2inch$. Provisional values as defined in ITU-T G.1070 are used for the rest of the parameters including coefficients $v3$, $v4$ and $v5$. In Figure 5, we can observe that the perceived video quality as defined by ITU-T G.1070 is greatly improved by the erasure codes compared to the standard CMT-SCTP when there is packet loss in the network.

We have also analysed the video application's play-out buffer evolution, *i.e.* the level of jitter introduced by the erasure coding mechanism and aRTX with respect to the lossy network paths. We compare the received jitter for standard CMT-SCTP and eCMT-SCTP in Figure 4. We can observe that jitter experienced by the video application notably decreases with the introduction of our erasure coding schemes due to early packet recovery without retransmission.

IV. PERFORMANCE VALIDATION

In this section, we present the performance validation of the fully reliable eCMT-SCTP by using the multi-homed SCTP analytical model from [21]. The estimated aggregated goodput is calculated as shown by the Equation 6 with respect to the

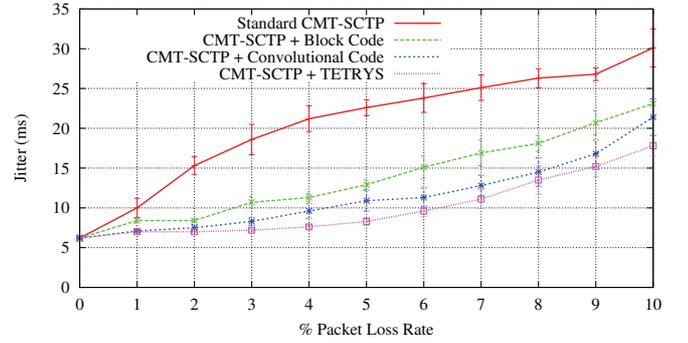


Figure 6. Jitter as experienced by the video playback application

total usable data capacity of $1800kbit/s$ and varying loss of 1% – 10% in our experiments. As can be seen in Equation 6, this model is function of packet loss expressed by p_j for each of the paths in a multi-path scenario. In Equation 6, W_j is the congestion window of the corresponding paths, RTT_j^r mean RTT for retransmission, RP_j is the mean number of packets transmitted via alternative path, W_j^r is the mean congestion window during retransmission and W_{Max}^j is the maximum window size which is the size of the receiver's window in our case. Q_j in Equation 6 is calculated as shown in Equation 7. RTT_j^r and W_j^r are calculated as Equation 8 and 9 respectively. Calculation of RP_j is shown by the Equation 10 and W_j is calculated as 11.

For selected erasure codes, we measure the corresponding packet loss rate at the decoder output in NS-2 experiments that utilise eCMT-SCTP with unreliable data transmission. This allows us to capture the true packet loss rate (PLR_{out}) in presence of the erasure codes as experienced by the transport layer. Figure 7 shows the input to output packet loss rates for block, convolutional and on the fly convolutional codes for a varying input packet loss rate of 1%–10%. We use these values in Equation 6 for eCMT-SCTP goodput calculation. Figure 8 shows the resulting eCMT-SCTP throughput estimates, compared to the NS-2 simulation results for fully reliable eCMT-SCTP. We only show a subset of the results, with modelling of aRTX retransmission mechanism left for future study. We can observe a closer match between the model and simulation results as the packet loss in the network increases, while a notable fixed mismatch remains in case of no and minimum packet loss. Our analysis shows that this mismatch is primarily contributed by the implications of receiver's buffer blocking and imprecise blind round-robin packet scheduling for CMT-SCTP in NS-2 simulations which the model in [21] doesn't take into account.

$$Q(w) = \min \left(1, \frac{\left(1 - (1-p)^4\right) \left[1 + (1-p)^4 \left[1 - (1-p)^{w-3}\right]\right]}{1 - (1-p)^w} \right) \quad (7)$$

$$B_j(p_j) = \min \left(\frac{W_{Max}^j}{RTT_j^r}, \frac{Q_j \times (1.5W_j - 2) + 1.25W_j + \frac{1-p_j}{p_j}}{Q_j \times \left[To + RTT_j^r \left(\frac{1}{2} + \frac{RP_j}{W_j^r} \right) \right] + RTT_j \left[Q_j \times \log 1.5 \left(\frac{W_j+2}{2} \right) + \frac{W_j-2}{2} \right]} \right) \quad (6)$$

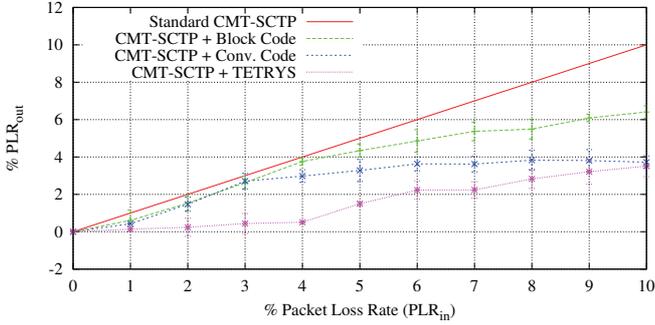


Figure 7. Packet error rate (PLR_{out}) experienced by CMT-SCTP

$$E[RTT_j^r] = \sum_{l=1, l \neq j}^N \frac{\frac{1}{p_l}}{\sum_{m=1, m \neq j}^N \frac{1}{p_m}} \times RTT_l \quad (8)$$

$$E[W_j^r] = \sum_{l=1, l \neq j}^N \frac{\frac{1}{p_l}}{\sum_{m=1, m \neq j}^N \frac{1}{p_m}} \times E[W_l] \quad (9)$$

$$RP(w) = \frac{(3-w).p^3 + (4w-11).p^2 + (14-6w).p + (4w-6)}{-p^3 + 4p^2 - 6p + 4} \quad (10)$$

$$E[W_j] = \frac{4}{3b} + \sqrt{\frac{8}{3bp} + \left(\frac{4}{3b}\right)^2} \quad (11)$$

V. RELATED WORK

In this section we present an overview of related work in both single and multi-path transport protocols in comparison to our contributions in this paper. In [7] Dihong et al. proposed a block code based packet error recovery scheme for SCTP called "Parity Streams" to improve performance of single and multi-stream data transfer with *single-path* SCTP. In comparison to [7], our proposal evaluates performance of not just block codes but also more error-resilient convolutional and on-the-fly erasure code for multipath CMT-SCTP protocol. In [15] Yong et al. proposed a fountain code [16] based scheme for multipath TCP called FMTCP, capable of transmitting different encoded data blocks over different paths and is resilient to path diversity as in varying RTT, loss, jitter and capacity. In [17] Y. Hwang presented another fountain code based scheme called HMTP. In HMTP, the encoded packets over multi-homed paths are transmitted until the receiver sends an acknowledgement back to the sender. This send-until-stop

transmission scheme makes HMTP to perform inefficiently and makes it prone to redundancy. In comparison to fountain code based multipath TCP derivatives, our work complements these contributions with block, convolutional and on-the-fly erasure code adaptable to both single and multipath based SCTP. This send-until-stop transmission scheme results in inefficient HMTP performance with an unnecessary level of included redundancy. In comparison to fountain code based multipath TCP based mechanisms, our work uses block, convolutional and on-the-fly erasure codes which are integrated within the multipath based CMT-SCTP. In [18] K. K. Lam et al. evaluated and demonstrated improvement for multipath TCP with Reed-Solomon erasure code. They concluded that using the erasure code improved TCP performance under stable channel conditions but failed to provide significant improvement under noisy scenarios. Authors concluded that the main reason for this is the use of TCP protocol, which acts as a bottleneck under noisy channel conditions. Finally, in [19] V. Sharma et al. proposed yet another modification of TCP called MPlot to mitigate bursty and correlated packet losses under wireless mesh network environments, using erasure codes and a hybrid ARQ/FEC scheme. MPlot aims to provide the best balance between goodput and packet delivery delay for the application layer.

VI. CONCLUSION AND FUTURE WORK

From the presented results above, our erasure coded schemes clearly start to demonstrate benefits for 2% – 3% packet loss in the network. Our adaptive RTX scheme helps improve the performance of evaluated erasure codes further both by improving overall goodput and reducing spuriously retransmitted duplicate data packets saving useful bandwidth. Also, memory constrained mobile devices should be able to benefit from this eCMT-SCTP scheme, as our proposal does not introduce any additional buffering but innovatively shares the already existing buffer within the transport protocol. We plan to extend this work by integrating context based variable redundancy for bursty network loss in asymmetric multipath networking scenarios. Due to space constraints, we omit the evaluation of fairness to competing flows. We note that using ECN is beneficial to fairness when standard and eCMT-SCTP are concurrently used.

VII. ACKNOWLEDGEMENT

This research work has been supported by funding from National ICT Australia (NICTA). NICTA is a research organization funded by Australian Government research initiatives through Australian Research Council (ARC).

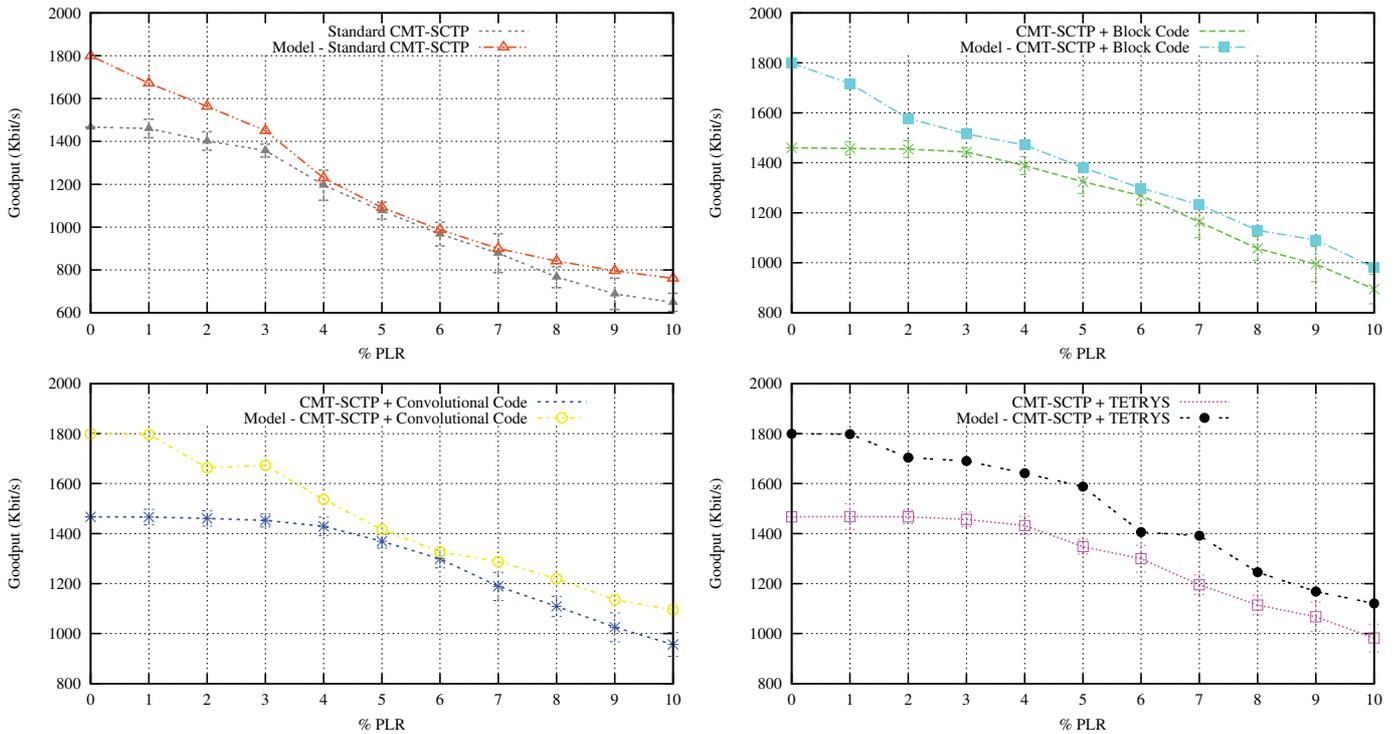


Figure 8. Performance validation of eCMT-SCTP

REFERENCES

- [1] IETF MPTCP Working Group, <http://datatracker.ietf.org/wg/mptcp>
- [2] Janardhan R. Iyengar, Paul D. Amer, Randall Stewart Concurrent Multipath Transfer Using SCTP Multihoming Over Independent End-to-End Paths, *IEEE/ACM Transactions on Networking*, 2006.
- [3] R. Stewart, Ed., RFC 4960 - Stream Control Transmission Protocol, 2007.
- [4] Mun Choon Chan, Ramachandran Ramjee, TCP/IP Performance over 3G Wireless Links with Rate and Delay Variation, *MobiCom 2008*.
- [5] Sinda Boussen, Nabil Tabbane, Sami Tabbane, Performance Analysis of SCTP Protocol in WiFi Network, *ICCIT 2009*.
- [6] R. Stewart, M. Tuexen, X. Dong, ECN for Stream Control Transmission Protocol (SCTP), *IETF Draft*, February 2012.
- [7] Dihong Tian, Ghassan Alregib, Parity Streams - A Novel FEC Scheme with the Stream Control Transmission Protocol, *IEEE Communications Letters*, June 2006.
- [8] Jay Kumar Sundararajan, Devavrat Shah, Muriel Médard, Michael Mitzenmacher, Joao Barros, Network coding meets TCP, *INFOCOM 2009*.
- [9] Golam Sarwar, Rokhsana Boreli, Emmanuel Lochin, Ahlem Mifdaoui, Performance Evaluation of Multipath Transport Protocol in Heterogeneous Network Environments, *IEEE ISCT 2012*.
- [10] Jacob Chakareski, Philip A. Chou, Application Layer Error Correction Coding for Rate-distortion Optimized Streaming to Wireless Clients, *IEEE Transactions on Communications*, 2004.
- [11] Thomas Stockhammer, Amin Shokrollahi, Mark Watson, Michael Luby, Tiago Gasiba, Application Layer Forward Error Correction for Mobile Multimedia Broadcasting, *Handbook of Mobile Broadcasting: DVB-H, DMB, ISDB-T and Media FLO*, March 2008, CRC Press, Taylor & Francis Group, Borko Furht, Syed Ahson.
- [12] David J.C. MacKay, *Information Theory, Inference, and Learning Algorithms*, ISBN-13: 9780521642989, March 2005.
- [13] Pierre-Ugo Tournoux, Emmanuel Lochin, Jerome Lacan, Amine Bouabdallah, Vincent Roca, Vincent On-the-fly erasure coding for real-time video applications, *IEEE Transactions on Multimedia*, vol. 13, pp.797-812.
- [14] Pierre-Ugo Tournoux, Emmanuel Lochin, Henrik Petander, Jérôme Lacan, A Packet Error Recovery Scheme for Vertical Handovers Mobility Management Protocols, *ICST MobiQuitous*, December 2010.
- [15] Yong Cui, Xin Wang, Hongyi Wang, Guangjin Pan, Yining Wang, FMTCP: A Fountain Code-Based Multipath Transmission Control Protocol, *IEEE ICDCS 2012*.
- [16] D.J.C MacKay, *Capacity Approaching Codes Design and Implementation Special Section*.
- [17] Younghak Hwang, Brownson O. Obele, and Hyuk Lim, Multipath Transport Protocol for Heterogeneous Multi-homing Networks
- [18] Ka Kit Lam et al., Performance Analysis and Optimization of Multipath TCP, *IEEE WCNC 2011*.
- [19] Vicky Sharma, Kalyanaraman Shivkumar, Kar Koushik, K. K. Ramakrishnan, and Vijay Subramanian, MPLoT: A Transport Protocol Exploiting Multipath Diversity using Erasure Codes, *INFOCOM 2008*.
- [20] The Network Simulator - NS-2, <http://www.isi.edu/nsnam/ns/>
- [21] Yi, Zhengliang, Tarek Saadawi, and Myung Lee, Analytical Modeling of Load Balancing SCTP. *New York Metro Area Networking Workshop (NYMAN)*, 2004.
- [22] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, P. Conrad, RFC 3758 - Stream Control Transmission Protocol (SCTP) Partial Reliability Extension, May 2004.
- [23] International Telecommunication Union, ITU-T Recommendation G.114 - One-way transmission time, May 2003.
- [24] Tuan Tran Thai, Emmanuel Lochin, Jérôme Lacan, Online multipath convolutional coding for real-time transmission, *IEEE 19th International Packet Video Workshop (PV)*, 2012.
- [25] International Telecommunication Union, ITU-T Recommendation G.1070 Opinion model for video- telephony applications, April 2007.
- [26] Q. Xie, R. R. Stewart, C. Sharp, I. Rytina, *IETF Draft - SCTP Unreliable Data Mode Extension*, April 2001.
- [27] Golam Sarwar, Rokhsana Boreli, Emmanuel Lochin, Ahlem Mifdaoui, Guillaume Smith, Mitigating Receiver's Buffer Blocking by Delay Aware Packet Scheduling in Multipath Data Transfer, *International Workshop on Protocols and Applications with Multi-Homing Support*, December 2012.
- [28] Thomas Dreiholz, Martin Becke, Erwin Paul Rathgeb, Michael Tüxen, On the Use of Concurrent Multipath Transfer over Asymmetric Paths, *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, December 2010.