VISIBLE: Virtual Congestion Control with Boost ACKs for Packet Level Steering in LWIP Networks

Thomas Valerrian Pasca, Bheemarjuna Reddy Tamma, and Antony Franklin.

Department of Computer Science and Engineering
Indian Institute of Technology - Hyderabad

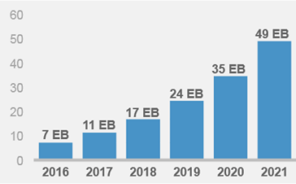
भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Outline

Introduction

- Cisco: Mobile data traffic will grow 7x by 2021 compared to that in 2016 [1]
- Mobile operators need to significantly improve network capacity to meet the user demand
- Utilizing unlicensed band efficiently has gained operators interest

- Wi-Fi offloading is a sweet spot for addressing the bandwidth crunch
- We focus on LWIP for harvesting the benefits of unlicensed band



Radio Level Interworking Architectures

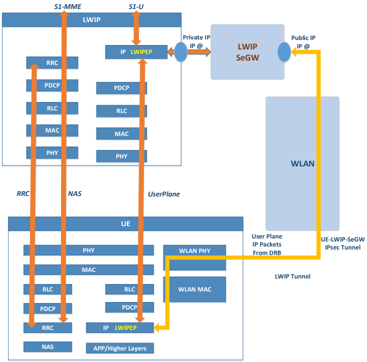


Figure : LWIP

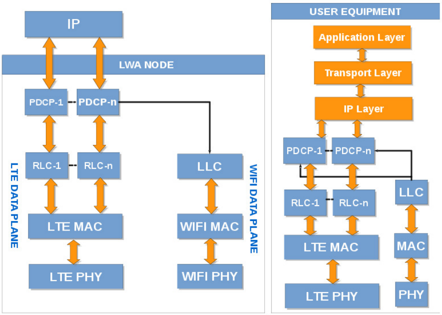


Figure : LWA

Motivation

- Tighter LTE and Wi-Fi interworking architectures can harvest maximum benefit of link aggregation with packet level steering
- Packet level steering may lead to Out-of-Order (OOO) delivery of packets at the receiver due to link heterogeneity
- TCP receiver generates DUPlicate ACKnowledgements (DUP-ACK) for OOO packets received
- The unnecessary DUP-ACKs adversely affect the TCP congestion window growth and then lead to poor TCP performance
- Efficient packet level steering and avoiding OOO delivery are necessary to reap in full benefits of LWIP

Literature Review

Major issues with packet level steering in LWIP can be classified under two major categories:

1. Problem with packet reordering
2. Spurious Retransmissions

Reducing Spurious Retransmissions

- Enabling TCP sender to precisely differentiate congestion loss from OOO packet delivery
- DOOR [5] detects OOO delivered packet by an additional ordering information in the ACK
- DOOR adds one byte TCP option field ACK Duplication Sequence Number (ADSN) into TCP ACK header
- Extension of DOOR, TCP-DOOR-TS [6] uses TCP timestamp mechanism
- Sender tracks the packet sending time and receiving time and relatively calculates the time stamp of every packet for detection of OOO packet.
- TCP receiver sets OOO option bit to inform the sender

Packet Reordering Techniques

1. Delayed ACK [8] introduces a waiting time before the receiver generates a DUP-ACK
2. Delay in ACK generation provides opportunity for the receiver to minimize the necessity for generating DUP-ACK
3. Delaying ACK in slow start phase will negatively affect TCP growth
4. Reordering Robust-TCP [9] and TCP-Packet Reordering [10] target to prevent persistent packet re-ordering from contrivedly activating congestion response by deferring packet retransmission and congestion response till the occurrence of packet loss

Packet Steering Techniques in LWIP

Lowest RTT First (L-RTT)

1. Employed as MPTCP's default scheduler
2. First fills the congestion window of the link with the lowest RTT and then the link with higher RTT.

Queue Depletion Rate (Q-Depl)

1. The rate of decrease in the length of the queue is used as a factor for steering the traffic
2. Queue depletion rate is comparable to available data rate of a radio link

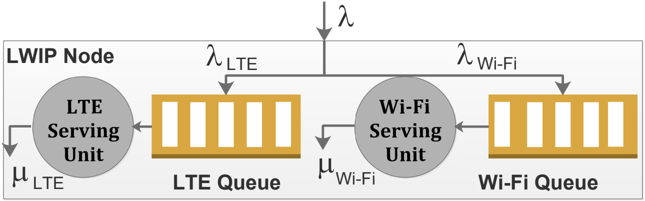


Figure : Packet steering model at LWIP node

Lowest RTT First:

$$L_{L-RTT} = \left(\frac{\lambda_{LTE}}{\mu_{LTE} - \lambda_{LTE}} - \frac{(N+1) \times (\frac{\lambda_{LTE}}{\mu_{LTE}})^{N+1}}{1 - (\frac{\lambda_{LTE}}{\mu_{LTE}})^{N+1}} \right) + \left(\frac{\lambda_{WiFi}}{\mu_{WiFi} - \lambda_{WiFi}} - \frac{(M+1) \times (\frac{\lambda_{WiFi}}{\mu_{WiFi}})^{M+1}}{1 - (\frac{\lambda_{WiFi}}{\mu_{WiFi}})^{M+1}} \right) \quad (1)$$

Queue with faster depletion rate:

$$\lambda_{LTE} = \frac{\mu_{LTE}}{\mu_{LTE} + \mu_{WiFi}} \times \lambda; \qquad \lambda_{WiFi} = \frac{\mu_{WiFi}}{\mu_{LTE} + \mu_{WiFi}} \times \lambda$$

Average queue length of L_{Q-Depl} based system is given by

$$L_{Q-depl} = \frac{2 \times \lambda}{\mu_{LTE} + \mu_{WiFi} - \lambda} - \left[\frac{(N+1)(\frac{\lambda}{\mu_{LTE}+\mu_{WiFi}})^{(N+1)}}{1 - (\frac{\lambda}{\mu_{LTE}+\mu_{WiFi}})^{(N+1)}} + \frac{(M+1)(\frac{\lambda}{\mu_{LTE}+\mu_{WiFi}})^{(M+1)}}{1 - (\frac{\lambda}{\mu_{LTE}+\mu_{WiFi}})^{(M+1)}} \right] \quad (2)$$

Comparison of different Packet Steering Techniques

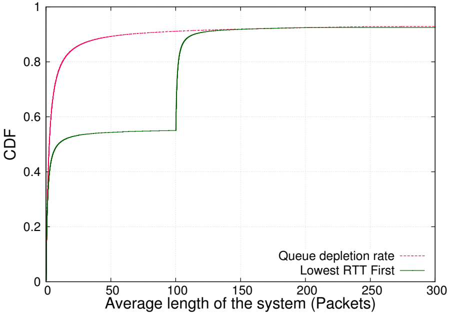


Figure : Queue-cdf

- CDF of average queue length of the system when λ is varied from 0 to $\mu_{LTE} + \mu_{WiFi}$
- L-RTT is filling the queue with lowest RTT which results in increase in its average queue length compared to Q-depl

Objectives of the Proposed Solution - VISIBLE

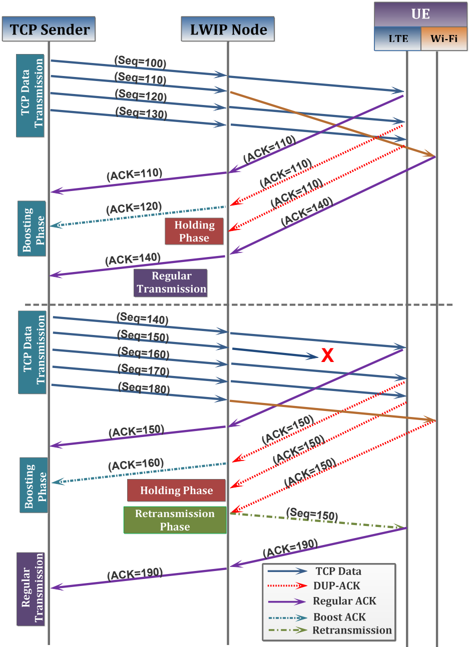
1. Enable growth of TCP congestion window
2. Reduce the triple DUP-ACK delivery to TCP sender
3. Support TCP packet retransmission at last hop

The three objectives are achieved through three different phases of the VISIBLE algorithm.

1. Boosting Phase
2. Holding Phase
3. Retransmission Phase

VIrtual congeStion control wIth Boost acknowLEdgement (VISIBLE)

- The objective includes High throughput, Reduction in OOO delivery and increased reliability
- Combination of packet steering technique and reordering technique to achieve higher throughputs
- No change to the TCP semantics
- No split in TCP session, Single congestion window at the sender


TCP Sender
LWIP Node
UE
LTE
Wi-Fi
TCP Data Transmission
(Seq=100)
(Seq=110)
(Seq=120)
(Seq=130)
(ACK=110)
(ACK=110)
(ACK=110)
(ACK=110)
(ACK=140)
(ACK=120)
Boosting Phase
Holding Phase
(ACK=140)
Regular Transmission
TCP Data Transmission
(Seq=140)
(Seq=150)
(Seq=160)
X
(Seq=170)
(Seq=180)
(ACK=150)
(ACK=150)
(ACK=150)
(ACK=150)
(ACK=160)
Boosting Phase
Holding Phase
Retransmission Phase
(Seq=150)
(ACK=190)
(ACK=190)
Regular Transmission
TCP Data
DUP-ACK
Regular ACK
Boost ACK
Retransmission

VISIBLE Algorithm I

TCP ACK Packet Received:

Update the TCP state information for all flows

if DUP-ACK of i^{th} flow && $\mathcal{A}_i^r > \mathcal{I}_i$ && $\mathcal{B}_{LTE}^a > \vartheta \times \mathcal{B}_{LTE}^s$ && $\mathcal{B}_{Wi-Fi}^a > \vartheta \times \mathcal{B}_{Wi-Fi}^s$ && $\varphi + \mathcal{P}_i < \frac{1}{\mathcal{N}} \times min(\frac{\mathcal{B}_{LTE}^a}{\mathcal{B}_{LTE}^s}, \frac{\mathcal{B}_{Wi-Fi}^a}{\mathcal{B}_{Wi-Fi}^s})$ && $\mathcal{RT}_i$==0 **then** ▷ Boost Acknowledgement Phase

$\mathcal{TH}_i \leftarrow \mathcal{CT}; \mathcal{H}_i \leftarrow 0; \mathcal{P}_i \leftarrow \mathcal{P}_i + 1$

$\mathcal{A}_i^r \leftarrow \mathcal{A}_i^r + (MSS \times \mathcal{P}_i)$

$Modify\_ACK\_Number(Packet, \mathcal{A}_i^r); \mathcal{A}_i^s \leftarrow \mathcal{A}_i^r$

else if DUP-ACK of i^{th} flow && $\mathcal{RT}_i == 0$ && $\mathcal{A}_i^r > \mathcal{I}_i$ &&$\mathcal{P}_i < \mathcal{PH}_i^a \times min(\frac{\mathcal{B}_{LTE}^a}{\mathcal{B}_{LTE}^s}, \frac{\mathcal{B}_{Wi-Fi}^a}{\mathcal{B}_{Wi-Fi}^s})$ && $\mathcal{H}_i < (\frac{1}{\mathcal{N}} \times ((LTE_{LTT} + WiFi_{LTT})/2) \times min(\frac{\mathcal{B}_{LTE}^a}{\mathcal{B}_{LTE}^s}, \frac{\mathcal{B}_{Wi-Fi}^a}{\mathcal{B}_{Wi-Fi}^s})$ && $\mathcal{B}_{LTE}^a > \vartheta \times \mathcal{B}_{LTE}^s$ && $\mathcal{B}_{Wi-Fi}^a > \vartheta \times \mathcal{B}_{Wi-Fi}^s$ **then** ▷ Holding Acknowledgement Phase

if $\mathcal{TH}_i == 0$ **then**

$\mathcal{TH}_i \leftarrow \mathcal{CT}$

end if

$\mathcal{H}_i \leftarrow \mathcal{H}_i + \mathcal{CT} - \mathcal{TH}_i; \mathcal{TH}_i \leftarrow \mathcal{CT}; \mathcal{P}_i \leftarrow \mathcal{P}_i + 1$

return ▷ Stops the DUP-ACKs

else if DUP-ACK of i^{th} flow && $\mathcal{RT}_i < \mathcal{RT}^{max}$ && $\mathcal{B}_{LTE}^a > \vartheta \times \mathcal{B}_{LTE}^s$ && $\mathcal{B}_{Wi-Fi}^a > \vartheta \times \mathcal{B}_{Wi-Fi}^s$ **then** ▷ Retransmission Phase

$\mathcal{P}_i \leftarrow \mathcal{P}_i + 1; \mathcal{RT}_i \leftarrow \mathcal{RT}_i + 1$

$Trigger\_Local\_ReTx(\mathcal{A}_i^r, \mathcal{R}_j, \mathcal{BI}_{i,j})$

else ▷ Regular Transmission

VISIBLE Algorithm II

    if $\mathit{Get\_ACK\_Number}(\mathit{Packet}) == \mathcal{A}_i^r$ **then**
        $\mathcal{P}_i \leftarrow \mathcal{P}_i + 1$
    else
        if $\mathcal{P}_i > 0$ **then**
            if $\mathcal{P}_i > \mathcal{PH}_i^a$ **then**
                $\mathcal{PH}_i^a \leftarrow (1-\alpha) \times \mathcal{PH}_i^a + \alpha \times \mathcal{P}_i$
            else
                $\mathcal{PH}_i^a \leftarrow (1-\beta) \times \mathcal{PH}_i^a + \beta \times \mathcal{P}_i$
            end if
            $\mathcal{P}_i \leftarrow 0$
        end if
        $\mathcal{H}_i \leftarrow 0$; $\mathcal{TH}_i \leftarrow 0$; $\mathcal{RT}_i \leftarrow 0$;
        $\mathcal{A}_i^r \leftarrow \mathit{Get\_ACK\_Number}(\mathit{Packet})$
        $\mathcal{A}_i^r \leftarrow \mathit{Get\_ACK\_Number}(\mathit{Packet})$
    end if
end if
$\mathit{Send\_to\_S1U\_Socket}(\mathit{Packet})$

Experiment Setup

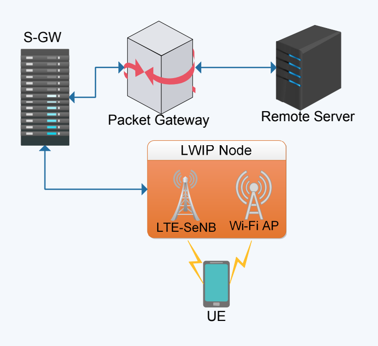


Figure : Experiment Setup

Table : EXPERIMENTAL PARAMETERS

Parameter	Value
Number of LWIP Node and UEs	1, [1 to 4]
LTE SeNB bandwidth	10 MHz, FDD
LTE and Wi-Fi Tx power	20, 16 dBm
LTE antenna model	Isotropic antenna model
LTE path loss model	Friis propagation loss model
LTE SeNB scheduler	Proportional fair
Wi-Fi frequency, bandwidth	2.4 GHz and 5 GHz, 20 MHz
Wi-Fi standard	IEEE 802.11 a/b/g
Wi-Fi rate control algorithm	Adaptive auto rate fallback
Application	TCP BulkSend Application
TCP congestion control algorithm	TCP New Reno
Buffer size of LWIP Node	40 packets (per interface)

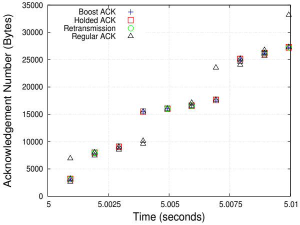


Figure : Different phases of VISIBLE

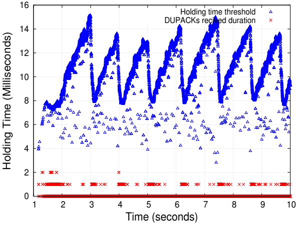


Figure : Holding time of ACK packets

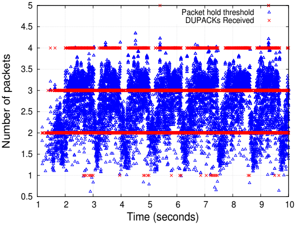


Figure : Number of DUP-ACK held

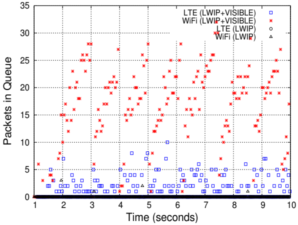


Figure : Lengths of LTE and Wi-Fi queues

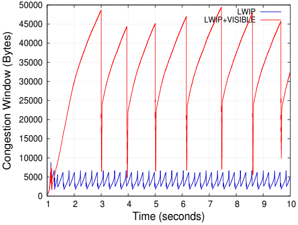


Figure : Congestion window behaviour

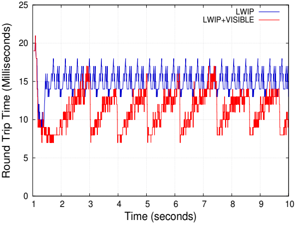


Figure : RTT of TCP flows observed

Performance Comparison with VISIBLE Algorithm

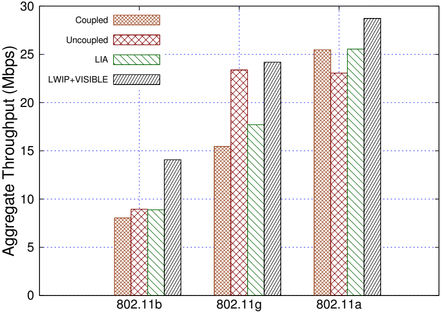


Figure : Throughputs of MPTCP and LWIP+VISIBLE

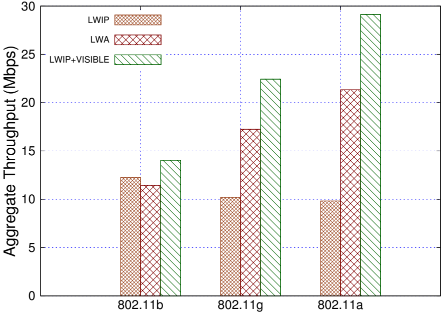


Figure : Throughputs of LWA, LWIP, and LWIP+VISIBLE

Performance comparison with MPTCP

1. Two downlink flows are generated between RS and UE in full mesh mode of MPTCP
2. Various congestion control algorithms of MPTCP viz., Coupled, Uncoupled and Link Increase Algorithm (LIA)
3. When LTE and Wi-Fi link rates are incomparable, then MPTCP suffers from "the speed of the slowest link" problem and hence fails to achieve the aggregated throughput.
4. LWIP+VISIBLE has improved the throughput due to its Boost ACKs mechanism
5. When IEEE 802.11g is used (here LTE and Wi-Fi link rates are comparable), then MPTCP gets the full aggregation benefit
6. LWIP+VISIBLE also achieves comparable performance with MPTCP. When IEEE 802.11a is used, then LWIP+VISIBLE improves network throughput by 12% as compared to MPTCP

Conclusions

- VISIBLE has successfully aggregated multiple links even if they are of incomparable rates
- LWIP node incorporated packet steering technique based on queue depletion rate
- VISIBLE supports growth of congestion window of the sender to grow by sending Boost ACKs in a controlled fashion from LWIP node
- VISIBLE has out performed MPTCP based LTE-Wi-Fi integration by 37% and LWA architecture by 30%

Acknowledgements

This work was supported by the project "Converged Cloud Communication Technologies"

सत्यमेव जयते

Ministry of Electronics and Information Technology
Government of India

References I

[1] Cisco. (2017) A White Paper on Global Mobile Data Traffic Forecast Update. [Online]. Available: https://goo.gl/zY4nKI
[2] 3GPP, "LTE-WLAN Aggregation and RAN Controlled LTE-WLAN Interworking," Tech. Rep. 36.300, 2016.
[3] P. Thomas et al., "Architectural challenges and solutions for collocated lwip: Network layer perspective," in Proc. of NCC. IEEE, March 2017.
[4] 3GPP. (2015) Study on Small Cell enhancements for E-UTRA.
[5] F. Wang and Y. Zhang, "Improving tcp performance over mobile adhoc networks with out-of-order detection and response," in Proc. of MobiHoc. ACM, 2002, pp. 217-225.
[6] H. Mulugeta et al., "Performance improvement of tcp using tcp-door-ts algorithm in mobile ad hoc networks," in Proc. of ICCT. IEEE, 2011.

References II

[7] R. Ludwig et al., "The eifel algorithm: making tcp robust against spurious retransmissions," ACM SIGCOMM CCR, vol. 30, no. 1, 2000.
[8] V. Paxson, "End-to-end internet packet dynamics," in ACM SIGCOMM CCR, vol. 27, no. 4, 1997, pp. 139-152.
[9] M. Zhang, B. Karp, S. Floyd, and L. Peterson, "RR-TCP: a reordering robust TCP with DSACK," in Proc. of ICNP. IEEE, Nov 2003.
[10] S. Bohacek et al., "A new tcp for persistent packet reordering," IEEE/ACM Transactions on Networking, vol. 14, no. 2, April 2006.
[11] Y. Khadraoui et al., "Tcp performance for practical implementation of very tight coupling between lte and wifi," in Proc. of VTC. IEEE, Sept 2016, pp. 1-6.
[12] MPTCP. [Online]. Available: www.github.com/mkheirkhah/mptcp


THANK YOU
MERCI
OBRIGADO
ASANTE
СПАСИБО
SPASIBO
WELALIN
ZIKOMO
MULTUMESC
THANK YOU VERY MUCH
どうも
SUWUN
EIO
MAAKE
MATONDO
БАЯРЛАЛАА
MOCHCHAKKERAM
МНОГО ВИ БЛАГОДАРЯ
SAAGHA XALDA HWA
RAIBH MAITH AGAT
KHUBLEI
VINAKA
QUYANAQ
CHOKRANE
GRAZIE
MATUR NUWUN

For Further Queries Contact US

cs13p1002@iith.ac.in

www.thomasvalerrianpasca.in

GitHub https://github.com/ThomasValerrianPasca/