ML Assisted Feedback Mechanism for TCP Congestion Control in Next Generation Wireless Networks

Vasanth Kanakaraj, Shreyanshu Agarwal, Keerthi Priya P, Vishal Murgai and Issaac Komminenni
Networks S/W R&D Group, Samsung R&D Institute, Bengaluru, India

2022 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS) | 978-1-6654-7340-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ANTS56424.2022.10227666

***Abstract*—5G and beyond delivers high data throughput, in turn, requiring proactive mechanism in core and Radio Access Network (RAN) fast path for delivering congestion feedback to Congestion Control (CC) algorithm of Transmission Control Protocol (TCP). Mobile handovers between cells with higher variance in Bandwidth Delay Product (BDP) especially in a heterogeneous network present multitude of challenges for transport layer protocols such as TCP and QUIC. Inability of existing CC mechanisms in TCP and QUIC to adjust to sudden changes in BDP result in packet queueing delays, packet drops leading to degraded user experience. In this paper, we propose a novel method to implement feedback mechanism in RAN fast path using Machine Learning (ML) model to predict impending congestion event and notify TCP's CC. This paper also discuss about implementation of a proactive bandwidth regulation function in User Plane Function (UPF) by using ML to predict an optimal maximum TCP Receive Window (RWND) size. We demonstrate the effectiveness of our proposed method in both RAN and core network with Key Performance Indicators (KPIs) collected from live air network. The results from conducted experiments demonstrate 17% reduction in Flow Completion Time (FCT) and 15% reduction in packet loss end to end.**

***Index Terms*—Machine Learning, TCP Congestion Control, 5G, RAN, Core Network, Neural Network**

I. INTRODUCTION

5G technology is designed to deliver large volumes of data with low latency and high reliability. The problem of congestion build up in next generation wireless networks is bound to increase due to the global adaptation of data intensive technologies and rapidly increasing number of wireless devices. TCP is most widely used transport layer protocol. TCP modulates the data transmission rate based on congestion control and flow control. Many Congestion Control (CC) algorithms running in User Equipment (UE), internet server are either loss or delay based. They tend to react poorly when a sudden congestion event occurs. We propose a two pronged design that aims to implement proactive congestion notification for end devices in Radio Access Network (RAN) and enhance TCP flow control in core network. At RAN, we propose a proactive congestion feedback to CC of end device whenever there is a impending congestion event and congestion build up in RAN. Meanwhile in core network, we propose a method to regulate bandwidth per flow in 5G core's User Plane Function (UPF) as UPF is the key node connecting fast path of core with internet. In RAN, the main bottleneck for UE occurs during handover from high Bandwidth Delay Product (BDP) cell to low BDP cell as shown in Fig. 1 and during congestion buildup in the current cell of UE. It's tedious for endpoint's CC to predict the former case . Hence we propose a Machine Learning (ML) based intelligent mechanism to predict such UE handover event from high BDP to low BDP cell and end device will be notified about the impending congestion event through Explicit Congestion Notification (ECN).

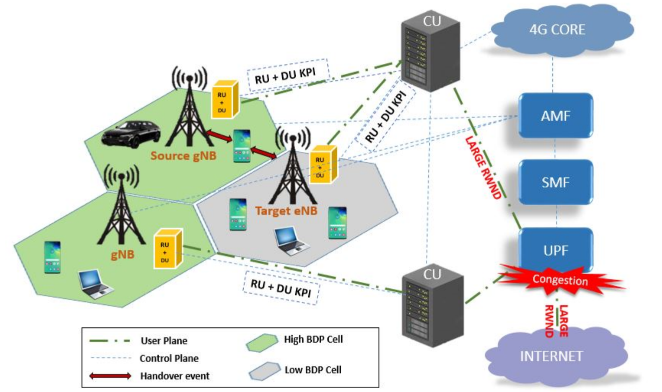


Fig. 1: Problem Statement

The TCP window scaling options enable a single TCP connection to set a maximum window size of 1GB (approx.). This window scaling parameter is negotiated during connection setup and it determines the amount of in-flight data between the UE and internet server. Data often flows from internet server to 5G UEs. This combined with larger memory available in 5G UEs can often lead to higher TCP Receive Window (RWND) advertised by UEs during connection establishment. TCP in internet server estimates congestion in the network for limiting the number of in-flight packets to UE. Since 5G delivers good throughput with improved reliability, often the advertised RWND from UE will be less than the server estimated Congestion Window (CWND). In this case, the server's TCP will transmit data at advertised RWND rate.

Similar reactions from many such internet servers can overwhelm the operator network's fast path with traffic bursts creating scenarios for packet buffering, packet loss and congestion in UPF as shown in Fig. 1. There is ongoing work for fixing an upper bound for RWND size advertised by UE. This motivated us to propose an efficient method using ML to predict optimal maxRWND for bandwidth regulation in UPF. The proposed two pronged design has been studied extensively in Section

III and IV. Section V describes performance evaluation of proposed method in detail.

II. RELATED WORK

This section provides an overview of existing literature work aimed at mitigating congestion build up for TCP flows in core as well as RAN. Menglei Zhang et al. [1] emphasizes challenges emerging with TCP's adaptability to mmWave 5G cellular networks. TCP's selective acknowledgment (SACK) option to avoid multiple packet losses, Active Queue Management (AQM) schemes to control the behaviour of queues and buffers, proxying, link layer retransmissions between gNodeB (gNB) and the UE are the different TCP performance enhancement Techniques mentioned. Our paper aims at addressing the above mentioned unexpected congestion challenges that can happen during handover from high BDP to low BDP. Hadeel Abdah et al. [2] proposed Probabilistic and Enhanced Handover Prediction Approach based on global users movement history and Individual profiling. Probabilistic Handover Prediction approach derives the probabilities of handover to base station from the user's handover history given the user's previous cell at each base station and based on these probabilities, and arrives at the handover decision. But [2] doesn't consider the time UE's Congestion Control Algorithm takes to adapt to the prevailing congestion in the network. Our proposed handover ML model predicts the handover 800 ms in advance so the ECN bit set packets reach the UE and the UE would be able to adapt to congestion in due time.

In [3], the authors propose an architecture for RWND modulation based on Round Trip Time (RTT) per flow along with available line capacity. This method relies on two edge routers that can actively regulate the RWND based on the capacity of line between them. The proposed solution [3] may not be optimal for 5G deployment due to presence of wireless medium that can cause multiple variations in path and RTT. This makes RTT computation between endpoints a more tedious process and also, the gNB has a high bandwidth, good quality connection with core network. This mitigates the necessity of two edge nodes to regulate the advertised RWND. Authors in [4] have studied the impact of mobility on the advertised RWND of TCP. They have analysed in detail regarding how different types of congestion control algorithms (loss and delay based) in TCP react to path characteristic changes caused by mobility and its effect on RTT, throughput. We have suggested a different approach that is related to mitigating congestion with the help of the flow control in TCP. Google has published a new congestion control algorithm called BBR [5]. This aims to solve the performance issues caused by loss and delay based congestion control algorithms that are widely in use. But BBR is found to be unfair with respect to flows having different RTT and its performance can fluctuate in case of paths having smaller buffers. In our proposed approach, a uniform maxRWND is set as upper bound on all TCP flows (with same QoS) to ensure fairness and also, our paper suggests a method to mitigate congestion by regulating flow control. Authors in [6] also suggest modification of advertised RWND size as a means for congestion mitigation. They tested TCP/IP connections over Asynchronous Transfer Mode (ATM) network. In this paper, the advertised RWND is a function of available buffer space in ATM router and benefits of the same has been analysed in detail. But our proposed solution is tested on a prevailing wireless environment and also, the ATM has been superseded by IP networks. We have in turn considered multiple Key Performance Indicators (KPI) and implemented ML model for predicting optimal maxRWND for bandwidth regulation at UPF instead of considering the available buffer capacity alone as done in [6]. Other works in past [7]–[9] that propose improvements to the congestion control algorithms in TCP that would enable the sender adapt better for congestion in network. All these works require modification to TCP stack. Since TCP is most widely used transport layer protocol and any changes to TCP congestion control will require a huge effort from multiple vendors, platforms to adapt. Meanwhile, our proposed solution is independent of TCP sender's congestion control and it doesn't require any modifications to TCP stack.

III. PROPOSED DESIGN

This section provides an overview of our proposed two pronged approach to aid TCP endpoint with congestion and flow control. We demonstrate the necessity for a machine learning based solution for both RAN and core network's fast path. Our Proposed Method can be broadly categorized into two sections:

A. Proactive Feedback Method for Congestion in RAN

In 5G RAN, we propose a method to deliver proactive congestion feedback to endpoint's CC by estimating congestion build up in RAN (refer Algo. 1) and predict impending congestion event due to handover (refer Algo. 2). One Centralized Unit (CU) manages multiple Radio Unit (RU) and Distributed Unit (DU). RUs are co-located along with DUs and host many cells. CU collates wireless network KPIs. We have identified

Algorithm 1 Congestion estimation with feedback in RAN

for each $cell_i \in CU_j$ after $cell\_load\_interval$ **do**
    Compute $cell\_load_i$
    $state_{old} \leftarrow \text{state}_i$
    if $state_i = UNCONGESTED$ **and** $cell\_load_i < Thresh_{max}$ **then**
        Set $state_i$ to $CONGESTED$
    else if $state_i = CONGESTED$ **and** $cell\_load_i > Thresh_{min}$ **then**
        Set $state_i$ to $UNCONGESTED$
    end if
    if $state_{old} \neq state_i$ **and** $state_i = CONGESTED$ **then**
        CU_j sets the ECN bit for all packets belonging to the $cell_i$
    end if
end for

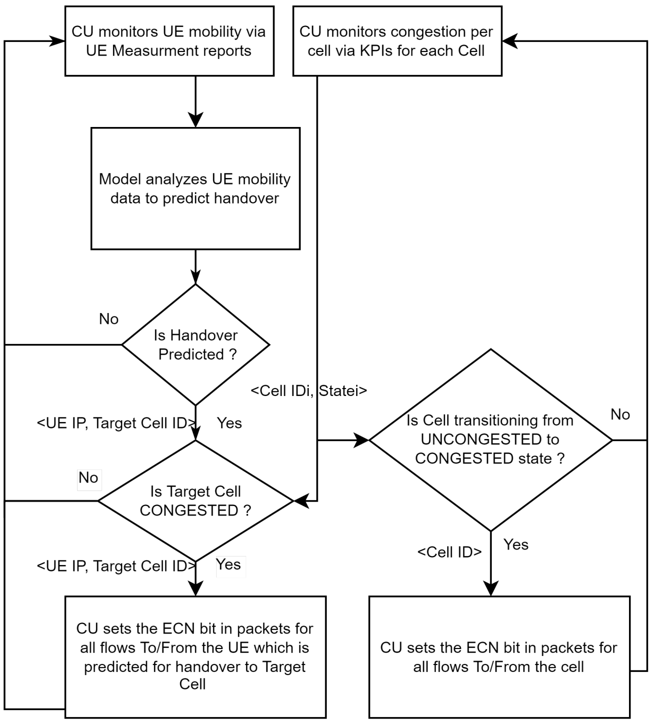


Fig. 2: Flowchart for Proposed congestion feedback mechanism in RAN

KPIs that can be used to estimate congestion per cell in each CU. This congestion data for all cells managed by CU will be shared with other neighboring CUs and vice versa. This in turn provides a collective intelligence about congestion (neighboring cells) in RAN.

We conducted a thorough study of KPIs in live network under different traffic load patterns to capture congestion scenarios in RAN. The study identified a set of KPIs that influence congestion per cell. The identified KPIs are average CPU and memory utilization of RU+DU, average queue utilization per QoS, packet loss rate, Packet Data Convergence Protocol (PDCP) downlink packet discard rate, load histogram and session setup success rate. These KPIs are easily accessible, monitored regularly at CU and play a major role in estimating congestion.

At every *Cell Load Interval* (configurable), cell state is determined as *Congested* or *Uncongested* (refer Algo. 1) maintained using congestion bit. This bit is set to 1, if cell is congested for the interval and 0 otherwise. Along with the collated congestion data per cell, user mobility per cell is also monitored by CU via UE measurement reports regularly. We determined that the UE measurement reports can be used to predict handover between cells as they play a major role in the handover decision in different handover algorithms. Moreover, they can be considered as a the most optimal method to track approximate user mobility as UE measurement reports are predominantly affected by user movement. Following which we pre-processed the dataset containing UE Measurement reports from over 7000+ cells obtained from live network and used it to train the neural network based Handover Prediction

Algorithm 2 ML Based UE Handover prediction in CU

```
1: Load pre-trained model weights and feature scalers
2: Initialize KPI ( UE Measurement Reports) Monitor on RU+DU
3: while True do
4:     Initialize features x_t using KPI data collected from RU+DU
5:     Normalize the values of each feature in x_t into proper ranges for prediction.
6:     Predict Cell_target for the UE using loaded model and pre-processed features in x_t obtained after 5.
7:     Handover prediction algorithm communicates the UE IP and Cell_target ID to the CU.
8:     if Cell_target is CONGESTED then
9:         ECN bit is set for packets belonging to the UE predicted to handover to Congested Cell.
10:    end if
11: end while
```

Model which takes the UE measurement report values across all neighbouring cells and predicts the target cell.

The handover prediction model deployed at CU predicts the target cell Id to which the UE will be handed over. If the target cell is in Congested state (refer Fig. 2) , then ECN bit is set for all TCP flows originating or terminating for the predicted UE. In our proposed method, the CU monitors congestion per cell and whenever a cell is transitioning from *Uncongested* to *Congested* state, ECN bit will be set for packets belonging to all UEs in that cell. ECN acts as the proactive feedback which enables TCP sender to react to the forthcoming congestion events in RAN. ECN bit is set only against UE handover to a congested cell, thereby minimizing the total UE handover decisions. Moreover, an audit mechanism is deployed to gauge the performance of the model in field and facilitate necessary manual interventions. Audit mechanism reverses the decision of setting ECN bit packets in case of false positives while false negatives results in momentary throttling.

B. Enhanced Flow Control for TCP in Core Network

Session Management Function (SMF) manages multiple UPFs of varying bandwidth capacity [10] and has access to all the KPIs of UPF. In UPF, factors like high queue utilization, high processor load due to traffic bursts, can lead to packet loss and high variation in RTT causing sudden congestion in UPF. We started identifying the KPIs that influence the bandwidth capacity of UPF. The identified KPIs are total number of active vs. idle users (current, peak, average), load of network processor (peak, average) which performs fast path function, user plane traffic load (peak, average).

In live network, the KPI study suggested that regulation of TCP bandwidth in UPF using a static maxRWND calculated based on the available processing capacity will not be an optimal solution due to the dynamically varying nature of user traffic patterns passing through UPF at any given time. Using a threshold based static maxRWND can lead to poor resource

Fig. 3: Variation of Data Traffic Load with Total Active Users

utilization and unnecessarily throttles user bandwidth during off peak hours. The number of active versus idle users keeps varying based on time of the day (peak time, off-peak time), day of the week (weekday, weekend, holiday). This in turn influences the traffic load in UPF as captured in Fig. 3. Also, there is no established pattern on rate at which active users became idle users and vice versa. Such dynamic changes in traffic patterns with respect to different KPIs necessitate a ML based prediction for an optimal maxRWND per TCP flow in UPF. SMF is best suited for deployment of the optimal maxRWND prediction model.

The window scaling parameter negotiated between the UE and the internet server during TCP connection establishment is stored by software block called flow manager. One SMF can manage n number of UPFs. In (1), $RWND_j$ represents an optimal maxRWND value per flow that is estimated for the i^{th} UPF serving N flows out of which k are TCP flows. CAP_i represents the bandwidth capacity of the i^{th} UPF. Then, the inequality in (1) holds true for all the UPFs:

$$\frac{\sum_{j=1}^{k} RWND_j}{CAP_i \times \frac{k}{N}} \leq 1 \tag{1}$$

In (2), $CAPM_i$ represents the memory capacity of the i^{th} UPF, and $CAPC_i$ represents the computational capacity of the i^{th} UPF.

$$CAP_i = CAPM_i + CAPC_i \tag{2}$$

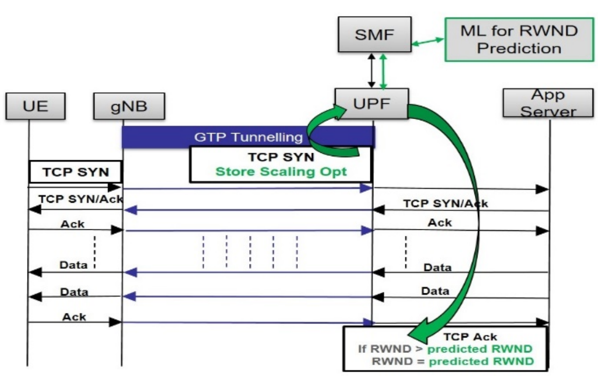


Fig. 4: Diagram for TCP maxRWND Prediction Model in 5G

In the proposed solution, the sum of the RWND values of all TCP flows going through UPF should be less than the total TCP bandwidth capacity of UPF in order to avoid congestion as explained in (1).

Algorithm 3 : ML Based TCP maxRWND Prediction Model

1: Initialize ML Model pre-trained using KPIs
2: Initialize KPI (Time of Day, Load per User Plane CPU(Average, Peak) UPF Traffic (Average, peak)) Monitor on SMF.
3: Initialize TCP maxRWND Prediction Interval
   $predInterval \leftarrow$ configured_prediction_$interval$ $seconds$
4: Initialize prediction timer $t \leftarrow 0$
5: **while** $True$ **do**
6:     **if** $t = 0$ **then**
7:         Initialize features x_t using KPI data collected by SMF
8:         Normalize the values of each feature in x_t into proper ranges for prediction.
9:         Predict $maxRWND_{predicted}$ using loaded model and pre-processed features in x_t obtained after 8.
10:         SMF communicates the $maxRWND_{predicted}$ to UPF.
11:         The UPF intercepts the TCP ACK packet
12:         **if** $RWND_{advertised} > maxRWND_{predicted}$ **then**
13:             UPF modifies the $RWND_{advertised}$ with $maxRWND_{predicted}$ in the TCP ACK packet
14:         **end if**
15:     **else if** $t = predInterval$ **then**
16:         $t \leftarrow 0$
17:     **else**
18:         $t \leftarrow t + 1$
19:     **end if**
20: **end while**

Once the ML model is trained and deployed in SMF, it will start predicting optimal maxRWND size per flow for UPF. SMF will then transfer the predicted maxRWND size to UPF as shown in Fig. 4. The frequency of maxRWND update will depend on the rate of change in KPI values for a monitored UPF.

UPF is capable of intercepting layer 4 contents in a packet. The flow manager in UPF will monitor all TCP flows for the RWND advertised by UE and Internet server. If advertised RWND size is more than the maxRWND, then the flow manager in UPF will replace advertised RWND value with predicted maxRWND value as described in Algo. 3. The flow manager will update the modified packet with recomputed TCP checksum before forwarding it. The reduced RWND (predicted maxRWND) will be a limiting factor for TCP senders whose CWND is higher than the received RWND. Using the proposed method, we are able to implement a better flow control for TCP in core network.

IV. System Model

A. Dataset

The collected dataset captures all variation of the selected KPIs for good generalization of trained ML model. The dataset was acquired from core and RAN handling live traffic that are situated at different locations. The identified KPIs were collected from these gateways on an hourly basis for time period of 30 days. We expanded the obtained dataset for modelling, using Gaussian distribution. We modified the Probability Density Function (PDF) and Cumulative Distribution Function (CDF) as demonstrated in (3), (4) respectively, to ensure that the calculated number cannot be outside the range of minimum and maximum for an input variable.

$$PDF(x) = \begin{cases} 0 & ; x < x_{min} \text{ or } x > x_{max} \\ \frac{1}{\sigma\sqrt{2}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})} & ; \text{otherwise} \end{cases} \quad (3)$$

$$CDF(x) = \begin{cases} 0 & ;\ x < x_{min} \\ \frac{1}{2}(1 + \mathit{erf}(\frac{x-\mu}{\sigma\sqrt{2}})) & ; x_{min} < x < x_{max} \\ 1 & ; otherwise \end{cases} \quad (4)$$

where, μ = mean, σ = standard deviation,
x_{min} = minimum value of the input variable,
x_{max} = maximum value of the input variable,
erf is the Gaussian error function defined in (5)

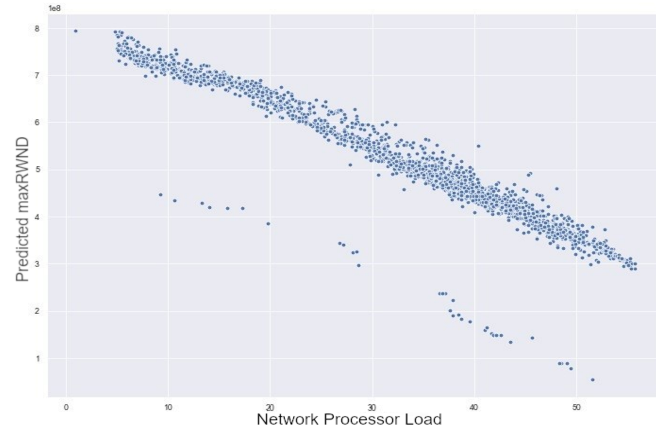
$$\mathit{erf}(z) = \frac{2}{\pi} \int_0^z e^{-t^2}\, dt \quad (5)$$

B. Prediction Modelling

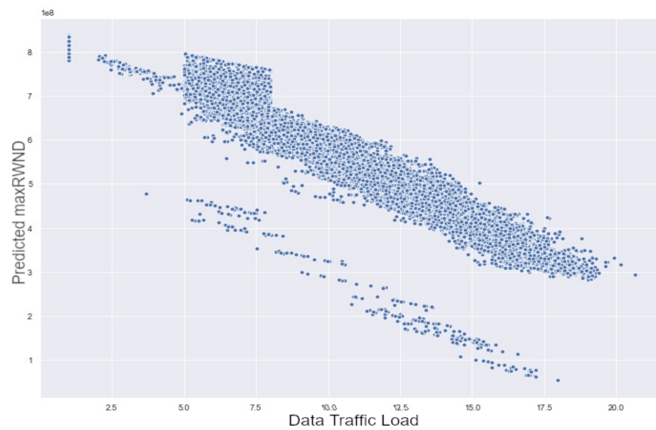
We use a feed forward neural network with two hidden layers and a dropout layer in between for handover prediction and optimal maxRWND prediction. In handover prediction model, the input layer consists of neurons corresponding to the each UE measurement report for current cell and neighbouring cells. The output layer consists of a single neuron corresponding to predicted Target Cell ID. Similarly, in maxRWND prediction model, the input layer consists of neurons corresponding to the each identified KPI and the output layer consists of a single neuron corresponding to predicted optimal maxRWND (size in MB). We did an exhaustive grid search to determine the optimum number of neurons in each hidden layer. To train the models, we used Adam optimizer [11] and Mean Squared Error (MSE) as the loss function.

V. Performance Evaluation

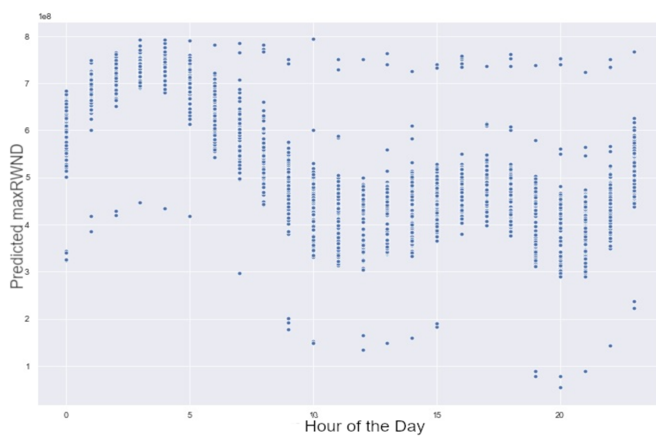
The proposed solution is evaluated on a testbed with 40,000 gNodeBs with over 1 million users against congestion and non congestion scenarios. We were able to achieve 15% less packet loss along with 17% reduction in Flow Completion Time (FCT). In the following section, we demonstrate the comprehensive analysis of our proposed method.



(a) Variation of Predicted maxRWND with Network Processor Load



(b) Variation of Predicted maxRWND with Data Traffic Load



(c) Variation of Predicted maxRWND with Hour of Day

Fig. 5: Variation of Predicted maxRWND with different KPIs

A. Analysis of Proposed Method in Core

We used 5-fold cross validation [12] for evaluating the performance of our maxRWND prediction model. The observed train and test loss values are 0.42 and 0.43 respectively, which demonstrates that the trained model is able to accurately predict an optimal maxRWND as per prevailing KPIs in UPF. Our model is able to predict maxRWND within $0.3 \sim 0.6$ ms. Our solution applies a uniform upper bound on maxRWND for

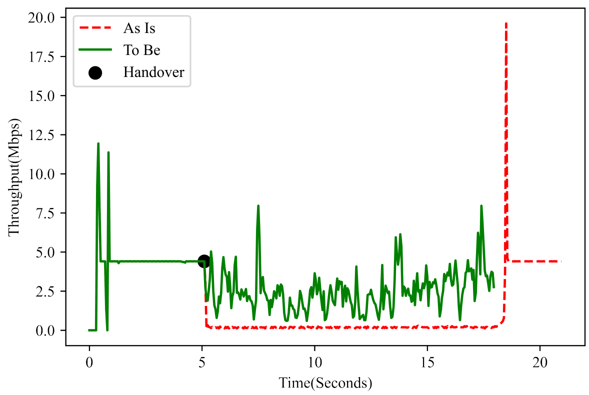


Fig. 6: Comparison of Throughput performance in handover scenario from high BDP to low BDP cell

all TCP flows in UPF at that instant, thus ensuring fairness for TCP flows in UPF's bandwidth share. The pairwise plots in Fig. 5 depicts the efficacy of the trained maxRWND prediction model on different variations of congestion scenarios. In Fig. 5a and 5b, we can observe a negative correlation between data traffic pattern, network processor's CPU load (that runs fast path software) and predicted maxRWND. Fig. 5c exhibits dynamic relationship between hour of the day and predicted RWND due to user traffic trends. It can be observed that traffic pattern changes based on time and type of the day (weekend, weekday, holiday). The results empirically prove the effectiveness of the proposed solution.

B. Analysis of Proposed Method in RAN

UE handover from high BDP to low BDP cell can lead CC to enter into slow start phase. This happens due to multiple packet losses as highlighted in Fig. 6 by the dashed red line (As Is). In proposed method, early feedback provided by the handover prediction model enables TCP's CC to enter into congestion avoidance phase as highlighted in Fig. 6 using green line (To Be). The handover prediction model predicts the handover event at least 800 ms before the actual handover occurs. The observed train and test loss values for the handover prediction model are 0.41 and 0.19 respectively which demonstrates that the ML model is able to predict accurately. Whenever there is a prediction of UE handover to low BDP cell, CU sets ECN bits in packets belonging to that UE. Since prediction happens 800 ms in advance, TCP and QUIC endpoints adapt to the events by regulating data sent into the network. This allows the applications in UE to adjust gracefully to the impending network conditions instead of reactive adjustment. The handover prediction model mitigates unexpected congestion during low BDP handover.

VI. Conclusion

This paper proposes a robust method using ML for enabling proactive congestion feedback along with enhanced flow control in 5G and beyond wireless networks. This has been verified with experimental results obtained from live air network. The collected live air statistics show that the TCP throughput makes 35% to 60% of total traffic in a user plane gateway. Hence, the proposed two pronged approach delivers a consistent TCP bandwidth performance along with an optimized link utilization. In core network, the proposed method enables optimal utilization of UPF's resources as the predicted maxRWND is based on the real time trend in KPIs. Early prediction of UE's Handover to a low BDP cell in RAN and marking packets with ECN bit in response to congestion build up proactively helps TCP's CC to scale proportionally to available bandwidth in RAN.

VII. Future Work

The study will be further expanded to further expand this research by enabling AQM for traffic shaping of fast path queues in RAN and core network. AQM in fast path will address better congestion avoidance for all transport protocols. In addition, we plan to include congestion awareness between RAN and core network for better congestion control. This will act as a holistic approach for mitigating congestion in wireless operator's network.

References

[1] M. Zhang, M. Polese, M. Mezzavilla, J. Zhu, S. Rangan, S. Panwar, and M. Zorzi, "Will tcp work in mmwave 5g cellular networks?" *IEEE Communications Magazine*, vol. 57, no. 1, pp. 65–71, 2019.

[2] H. Abdah, J. P. Barraca, and R. L. Aguiar, "Handover prediction integrated with service migration in 5g systems," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–7.

[3] F. Ciaccia, O. Arcas-Abella, D. Montero, I. Romero, R. Milito, R. Serral-Gracia, and M. Nemirovsky, "Improving tcp performance and reducing self-induced congestion with receive window modulation," in *2019 28th International Conference on Computer Communication and Networks (ICCCN)*, 2019, pp. 1–6.

[4] R. Mondal, A. Biswas, and B. Sardar, "Effect of mobility and receive window on tcp in device to device communication," in *2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, 2018, pp. 182–187.

[5] D. Scholz, B. Jaeger, L. Schwaighofer, D. Raumer, F. Geyer, and G. Carle, "Towards a deeper understanding of tcp bbr congestion control," *2018 IFIP Networking Conference (IFIP Networking) and Workshops*, pp. 1–9, 2018.

[6] L. Kalampoukas, A. Varma, and K. Ramakrishnan, "Explicit window adaptation: a method to enhance tcp performance," in *Proceedings. IEEE INFOCOM '98, the Conference on Computer Communications. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies. Gateway to the 21st Century (Cat. No.98*, vol. 1, 1998, pp. 242–251 vol.1.

[7] F. Keceli, I. Inan, and E. Ayanoglu, "Tcp ack congestion control and filtering for fairness provision in the uplink of ieee 802.11 infrastructure basic service set," in *2007 IEEE International Conference on Communications*, 2007, pp. 4512–4517.

[8] H. Wu, Z. Feng, C. Guo, and Y. Zhang, "Ictcp: Incast congestion control for tcp in data-center networks," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, p. 345–358, apr 2013. [Online]. Available: https://doi.org/10.1109/TNET.2012.2197411

[9] Y. Matsushita, T. Matsuda, and M. Yamamoto, "Tcp congestion control with ack-pacing for vertical handover," in *IEEE Wireless Communications and Networking Conference, 2005*, vol. 3, 2005, pp. 1497–1502 Vol. 3.

[10] 3GPP, "System Architecture for the 5G System," 3rd Generation Partnership Project (3GPP), TS 23.501, 2018.

[11] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980

[12] J. D. Rodriguez, A. Perez, and J. A. Lozano, "Sensitivity analysis of k-fold cross validation in prediction error estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 569–575, 2010.