Metering Re-ECN: Performance Evaluation and its Applicability in Cellular Networks

Ying Zhang, Ingemar Johansson, Howard Green, Mallik Tatipamula
Ericsson Research

Abstract—The Re-ECN protocol is a recently proposed congestion notification scheme for IP networks. Building upon Explicit Congestion Notification (ECN), which marks packets instead of dropping them during congestion, Re-ECN requires the end users to re-insert the marking information back to the network as a feedback to allow the network react more effectively on resource management. In this work, we first propose an infrastructure based deployment strategy by incorporating Re-ECN mechanisms into the GTP-U protocols in the cellular network without changing any end user terminals. We conduct simulation analysis of Re-ECN's performance in cellular network. We study the efficiency and fairness properties of Re-ECN in networks with a large number of short-lived and long-lived flows. A diverse set of network conditions is evaluated including different hops counts, variety of RTTs and loss rates. We further demonstrate its advantage by comparing with other congestion avoidance mechanisms.

I. INTRODUCTION

In spite of the economic downturn, mobile data traffic grows rapidly. The challenge wireless operators face is to support more subscribers with higher bandwidth requirements. To meet the bandwidth demand, operators need to seek for new technologies to efficiently utilize the available network resources, in particular, the resource allocation and flow management. Ample statistics for network traffic from the cellular network are available, stating that many short flows exist, but that a few large flows constitute a large part of the overall traffic volume [1]. Measurement studies have shown that a small number of users is responsible for the most part of the traffic in cellular network [2]. With the highly skewed network access behavior and more expensive resources in cellular network, resource allocation and usage accountability are the two important issues for the operators to achieve best network resource utilization. In particular, defining and enforcing resource fair sharing is the key problem we investigate in this work.

Existing proposals on fairness between multiplexed packet traffic, on both wired and wireless networks, are achieved by controlling relative flow rates. Flow rate fairness was the key objective behind widely deployed fair resource allocation protocols such as weighted fair queuing [3], TCP congestion control [4] and TCP-friendly rate control [5]. However, from both technical and economical point of view, equal flow rate does not imply fairness. The fairness should be applied to the principal entities in the network, *e.g.,* each user or groups of users. However, a flow is merely an information transfer between two applications. For instance, p2p applications exploit the vacuum of the flow-based fairness definition by just running more flows for longer periods. Moreover, because most operators today employ flat fee subscription model, allowing a single user to transfer tens of Gigabits per month, there is merely no penalty to these users economically. In cellular network, we are looking for a clean protocol architecture that performs a better job of automatically sharing out network capacity.

A recent proposal on "cost fairness" [6], [7] or Re-ECN [8], [9] introduces that fairness should be defined on one's action on other, *i.e.,* how much each user's transfers restrict other transfers, given capacity constraints. The metric should be the volume of congestion taken over time, more precisely, the congestion times the bit rate of each user causing it. In practice, it can be achieved by measuring the amount of packets dropped or marked during congestion. Importantly, unlike flow rates, this metric integrates easily and correctly across different flows on different paths and across time, so it can be easily incorporated into future charging models. Re-ECN [8], [9] builds upon ECN [10], a well known congestion avoidance scheme that marks packets instead of dropping them in face of congestion. The receivers return the marking information to the senders, who in turn decrease their transmit rate. In Re-ECN, the sender re-inserts the congestion information received back to the network as a feedback. Thus, as the packet passes by each router along the path, it carries a truthful prediction about the congestion condition of the remainder of their path. In other words, Re-ECN scheme provides sufficient information to hold senders and the whole network accountable for the congestion they cause downstream.

Re-ECN's capability of ensuring fairness could be used for ensuring fairness in cellular networks. However, although the idea looks promising, its advantages and performance have been not yet been verified thoroughly yet. In the context of cellular networks, which has more expensive resources and more stringent QoS requirements, the feasibility of applying Re-ECN to cellular network as well as its performance and deployment scenarios need to be examined. For instance, the mobile network may encounter longer delay and higher loss rate but most of the flows are short-lived. To ensure it works correctly, Re-ECN requires both the end hosts and the network to cooperate. When deploying Re-ECN in the cellular network, where and how to deploy the scheme is the first question we investigate. We then study its gain in improving application performance.

The second concern of applying Re-ECN on cellular network is its deployment obstacles. Similar to other TCP proposals, Re-ECN requires modifying end hosts to send the

feedback information to the network. This requirement is difficult to satisfy in cellular network given the diverse set of mobile phone vendors, smart phones, laptop data cards, *etc.*. In designing the Re-ECN framework in cellular network, our goal is to investigate whether we could create a network architecture that could be deployed today, so we limit ourselves from making any changes to the user devices, and we require that legacy applications work unmodified. Our design explores a new split in the responsibilities between mobile network infrastructure nodes *e.g.,* base stations and gateways, and the underlying transport networks. We show that it is possible to support Re-ECN without touching the user equipments (UEs). Note that Re-ECN is to be seen as a proof of concept, which has led to the formation of the ConEx working group in IETF [11]. ConEx will standardize a protocol that is similar to Re-ECN. In the remaining of this paper, we use the terminology of *Re-ECN* but the results are applicable to *ConEx*.

In this paper we first propose two different deployment scenarios of Re-ECN on the cellular network. Unlike the traditional Internet forwarding packets at the IP level, the packets transferred in mobile backhaul and mobile core are handled via a diverse set of tunnels such as GTP (GPRS Tunneling Protocol). For instance, the packets between cellphones and the core gateway (PDN-GW) is tunneled through intermediate base stations and another gateway (Serving-GW). The tunneling method provides a large degree of flexibility and control to the network operator on deploying flexible congestion control schemes. Thus, we consider two Re-ECN deployment scenarios: the end-to-end deployment model from the user equipments (UEs) to the Internet web servers, and the infrastructure model from the base station (eNodeB) to the cellular gateway PDN-GW.

Furthermore, we conduct a simulation-based evaluation study on Re-ECN's performance in cellular network context. We first investigate Re-ECN's performance related to different parameter settings. A simple yet clear set of recommendations for parameter configuration can significantly help the deployment of the protocol. Our analysis is valuable to the best practice of parameter configuration for Re-ECN. To fully understand its performance and benefits, we conduct comprehensive evaluation of Re-ECN under different network conditions with varied hop counts, latency and congestion levels. More importantly, we examine its usefulness under two deployment strategies in the 3GPP Long Term Evolution (LTE) networks. Interestingly we found that Re-ECN does not have significant benefit under the environment of high transmission error rate. When encountering the packet loss induced by transmission error, Re-ECN clients reduce the sending rate as if it was an indication of suggestion. In such cases Re-ECN does not have obvious benefits. Overall, we conclude that deploying Re-ECN in cellular network has significant advantage when the network loss rate is below certain threshold. Under severe lossy condition, there is less benefit.

This paper is organized as follows: we introduce the Re-ECN protocol in Sec II. Sec III describes the Re-ECN deployment strategies in cellular network. Evaluation methodology is presented in Sec IV and performance evaluation is shown in Sec V. We finally conclude and discuss future work in Sec VII.

II. THE RE-ECN PROTOCOL AND ITS MODELING

ECN [10] is a congestion avoidance scheme that marks packets instead of dropping them. The receivers of marked packets should return the information about marked packets to the senders who decrease the transmit rate. Two bits in the IP protocol are assigned to the ECN field. The sender clears the field to 00 (Not-ECT) if either end host is not ECN-capable. Otherwise it indicates an ECN-capable transport (ECT) using either of the two code-points ECT(0) (10) or ECT(1) (01). ECN-capable queues probabilistically set this field to 11 if congestion is experienced (CE).

Building upon ECN, Re-ECN reveals congestion information so that users and networks can be held accountable for the congestion they cause, or allow to be caused. Re-ECN builds on ECN so we briefly recap the essentials of the ECN protocol [10]. Each sender declares the amount of congestion expected to be caused in the ECN fields in the packet header. Each router along the path marks CE bit in the packet to indicate the actual congestion the sender causes. If the amount of CE marked packet (actual congestion) is larger than the claimed one, the flow is classified as a *negative flow*. The notion of a negative metric arises because it is derived by subtracting real congestion level from the claimed. Likewise in positive flows the claimed congestion is more than the actual level. The Re-ECN framework defines a few network entities to ensure the correct behavior of marking as well as the punishment to the negative flows. We summarize it in Figure 2.

- **Source congestion control:** The sender will throttle its rate as downstream congestion increases, with some version of TCP-friendly congestion response.
- **Ingress Policer:** The Re-ECN protocol ensures packets carry the necessary information about their own expected downstream congestion so that an access router can deploy a policer at its ingress to check that the source is complying with whatever congestion control it should be using. This policing can be done on either per flow or a per customer basis. Instead of fully relying on the users to declare his congestion due share, Re-ECN relies on a policer on the ingress to ensure the senders do not declare arbitrarily large congestion share. A modified bulk token-bucket is maintained with the parameters such as the initial token level, the filling rate and the bucket depth. The policer is to stop aggressive flows/users at the ingress, thus avoiding wasting any downstream resources. Below we briefly describe how the policer works. The policer controls the rate of a TCP-compliance flow to be $B_i = k\frac{s}{T*\sqrt{p}}$, where B_i is the throughput of flow i, k is a constant with an upperbound of $\sqrt{\frac{3}{2}}$, s is the average packet size, T is the round trip time of the flow, and p is the packet loss rate. Let us define $N = \frac{1}{p}$, which is the

average number of packets between two positive packets. Thus, we have $B_i = ks\frac{\sqrt{N}}{T}$. To obtain the inter-marking time, we take the derivative and get $\Delta_i = \sqrt{N}\frac{T}{k}$. Note that only the positive packets will consume the token bucket. Each policer empties the bucket by the size of each arriving packet and fills it at a rate the above flow rate $B_i = \frac{ks}{T\sqrt{p}}$, which can be derived from the Re-ECN fields. When receiving a packet, the Policer subtracts the packet size s from the bucket and adds $\frac{ks\Delta t}{T\sqrt{p}}$, where Δt is the inter-packet duration. If the bucket empties, sanctions are applied to the flow. For instance, all future packets might be discarded. Thus, the depth of the bucket D controls the room that each flow can deviate from its expected throughput. It should be chosen to minimize the likelihood that innocent flows are subject to sanction. A larger D will lead to smaller false positives. The dropping probability pl_i for a policer with token rate r, initial bucket I and bucket depth D at time i: $pl_i = \frac{B_i - \frac{ks_i}{T\sqrt{p}}}{B_i}$.

- **Egress Dropper:** With a policer to monitor the declared downstream congestion to prevent aggressive sources, the source has a clear incentive to understate downstream congestion. Thus, we need a guard at the egress to prevent such understatement. We introduce a dropper at the last network egress, which drops packets in flows that become negative downstream congestion, *i.e.,* the congestion caused in reality is higher than that is claimed. As traffic leaves the last router before reaching the receiver, the fraction of positive packets in a flow should match the fraction of negative introduced by congestion marking, leaving a balance of zero. When positive, dropper takes no action, meaning that the source is slower than it has to. The router drops packet in the negative flow to ensure fairness. As traffic leaves the last network before the receiver, the fraction of positive octets in a flow should match the fraction of negative octets introduced by congestion marking, leaving a balance of zero. If it is less (a negative flow), it implies that the source is understating path congestion.
- **ECN Marker:** Finally, to encode the congestion level to the packet, we need ECN enabled routers which mark the packets whenever congestion occurs.

Under this framework, network operators can constrain the overall amount of congestion a user can cause. Moreover, it provides flexibility for applications to choose different congestion control mechanisms and allows novel charging regimes to evolve.

III. Re-ECN Framework in LTE Networks

In this section we first briefly introduce the background of cellular network. We focus on 3GPP Long Term Evolution (LTE) network as it is the next generation cellular network with high requirement on data rate and resource management. We then describe the two deployment strategies we propose to apply Re-ECN in LTE networks.
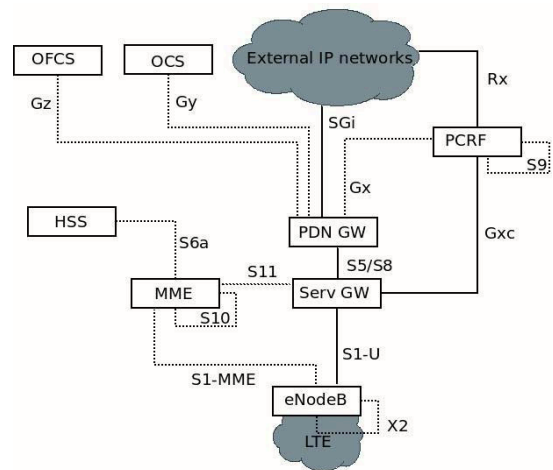

Fig. 1.   LTE Architecture

A. Introduction to LTE Cellular Network

Figure 1 depicts a typical LTE cellular network. In LTE network, the control level signalling traffic and the data traffic are handled separately. The user equipments (UEs) are connected to a base station, *i.e.,* eNodeB in LTE terminologies, via wireless interfaces. The eNodeBs transmit the control traffic and data traffic separately via two logically different interfaces. The control plane signaling is sent to the Mobility Management Entity, *i.e.,* MME, which MME handles all control functions. The user data payload, typically the IP packets, are sent to the Serving Gateway (Serving GW), and then subsequently to the Packet Data Network Gateway (PDN GW) in the core of the mobile network. The Serving GW tunnels all user data packets and buffers downlink IP packets destined for UEs that happen to be in idle mode. The PDN GW interconnects the mobile network to the external IP networks, *e.g.,* the Internet. The PDN GW includes functionality for IP address allocation, charging, packet filtering, and policy-based control of flows.

The data packets are not sent directly on the IP networks between the eNodeB and the GWs. Instead, every packet is sent in a tunneling protocol GPRS Tunneling Protocol (GTP) over UDP/IP. A GTP path is identified in each node with the IP address and a UDP port number on the eNodeB/GWs. GTP-tunnel is usually established per application for each user. The GTP protocol includes both for the data traffic (GTP-U tunnels) and the control traffic (GTP-C tunnels). This is very different from the end-to-end path on the Internet where the packet forwarding is performed at the IP level. Importantly, we observe that these tunneling protocols give the operator a large degree of flexibility to control the congestion mechanism incorporated with the GTP protocols. Below we will leverage this advantage to build the more *deployable* Re-ECN framework in LTE networks.
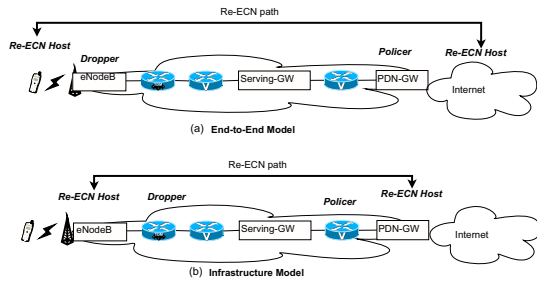
Proceedings of the 2011 23rd International Teletraffic Congress (ITC 2011)

Fig. 2.   Re-ECN Architecture Scenarios in LTE.

B. Re-ECN Deployment Strategies

Our first proposal of deployment is straightforward. The end-to-end path is defined conventionally to be between UEs and the web servers on the Internet for data packets and between the UEs and the MMEs for the control packets. In the following we focus on the data path due to its large traffic volume. In the following example, we illustrate the downlink direction which carries most traffic. Figure 2 (a) shows the deployment of the framework under this end-to-end definition. The *Policer* can be deployed on the first hop of the path under mobile operator's control, *i.e.,* the PDN GW or the first hop router connecting the PDN GW to the Internet. The *Dropper* is deployed at the egress of the path, in this example, the last hop in mobile backhaul, the eNodeB. It can also be deployed on the Serving-GW, depending on the resource contention concerns in the network.

The possible drawback of this deployment is its deployment overhead. It requires support by *e.g.,* the TCP stacks in the Operating Systems of interest, such as Linux,Windows on the web servers and Androids, Windows mobile on the phones. The sender needs to be able to insert the feedback back into the network. Routers along the path, especially on the bottleneck links should mark the ECN bit to indicate congestion. Moreover, it is common to have middleboxes such as firewalls, intrusion detection systems (IDS) set up along the path. Thus, the Re-ECN framework also requires that these middleboxes do not clear ECN marks or the Re-ECN information. This is hard to guarantee as the middleboxes may belong to different network management entities. For the above reasons it is reasonable to assume that deployment of Re-ECN that works end-to-end will take a long time.

The solution is to define Re-ECN locally. In this case Re-ECN is deployed along the GTP-U path between PDN-GW and eNodeB. This means that the Re-ECN loop only runs locally between PDN-GW and eNodeB. Neither ECN nor Re-ECN needs to be supported outside this path. The description below focuses on the download scenarios as the traffic volume on the downloading direction is much larger than upstreams. It is more likely to cause congestion due to the large traffic volume, in which case the infrastructure based Re-ECN is the most efficient. A similar setup can be done for upload scenarios.

Figure 2 (b) shows a different design: the end-to-end path

is defined between the eNodeB and the PDN-GW. In this case, Re-ECN is running on the traditional *IP* layer or *GTP* layer. Note that Re-ECN framework does not introduce any congestion control algorithms, rather it is a mechanism to expose information between UEs, gateways, and the network. The *Policer* can be deployed on the PDN-GW or the first hop router on the downstream after the PDN-GW. The *Dropper* can be on last hop before eNodeB or eNodeB itself. Moreover, if the path between the PDN-GW and the eNodeB is trusted, the *Dropper* is not required.

The steps to implement a localized infrastructure model Re-ECN solution are shown in Figure 3. In this case neither ECN nor Re-ECN is supported at the UEs, which is the most dominant cases in reality.
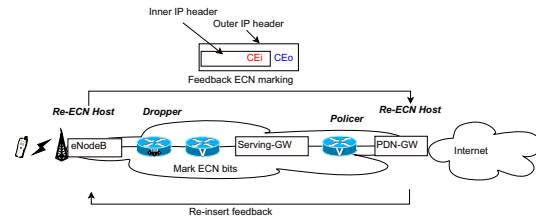

Fig. 3.   Infrastructure based Re-ECN framework

1) Supporting ECN in the GTP-U tunnels. The outer IP header of the GTP packet is indicated as ECN capable. The ECN-CE markings in the outer IP header should not be copied to the inner IP header at tunnel egress if the inner header is not ECN capable, which keeps the end user packets intact [12].The GTP-U protocol should use the optional sequence number to make the feedback reports reliable.

2) The outgoing interfaces on Serving-GW and PDN-GW are made ECN capable to set ECN-CE on the outer IP header probabilistically according to the average queue size. The routers along the path may also be ECN capable, *e.g.,* enabling ECN-RED which is widely supported in commercial routers.

3) eNodeB collects the ECN-CE marks and feeds it back to the PDN-GW. In this case, the eNodeB serves as the *receiver* in the normal Re-ECN/ECN deployment. This feedback contains the congestion occurred on the path between the eNodeB and the PDN-GW. However, the congestion can also happen on the eNodeB ingress queues or the Radio Link Control (RLC) level with AQM algorithms. This portion of congestion must also be fed back to the PDN-GW. Our idea to encode this congestion is to be logically *ORed* with the ECN-CE marks on outer IP header ($CE_i \| CE_o$ in Figure 3). The report format for the feedback can include the following information.

 • Sequence number of last received packets.
 • Total number of packets received by eNodeB.
 • Total number of packets dropped by eNodeB in the scheduler or the AQM.

- Total number of packets dropped by nodes before eNodeB, *i.e.,* GTP-U tunneled packets.
- Total number of packets ECN-CE marked by nodes before eNodeB, *i.e.,* GTP-U tunneled packets.
- List of marked and/or dropped packets, may be useful for computation of congestion volume, *i.e.,* sum of sizes of marked or dropped packets.

A typical report interval may be once every RTT or perhaps more seldom.

4) The PDN-GW gets the above congestion information in the report from the eNodeBs and can base on this information to find malicious flows, malicious users, aggressive sources of traffic, and also easily find overloaded spots in the network. The detection can be implemented by keeping the statistical value of ECN marked packets. PDN-GW can perform policing based on either per flow or a per customer basis. To ensure that a flow, a user, or a set of customers doesn't generate more congestion in the network than its due share, a modified bulk token-bucket can be maintained to allocate the resources gradually over time. The tokens represent the amount of congestion each user allows to consume in the network. Note that all traffic from a user over the lifetime of their subscription is policed in the same token bucket.

5) The PDN-GW re-inserts the congestion information into the GTP-U headers. Routers and S-GW along the path can prioritize the flows based on the feedbacks.

We named the second strategy the *Infrastructure Design*, which is easier to deploy as it can leave the end hosts unmodified. Moreover, the infrastructure nodes are owned by the same network operator entity. Thus, there is larger incentives for deployment to improve their own network performance.

The scalability of Re-ECN deployment is usually of concern. However, in LTE network, fine-grained flow level control is already adopted for other resource management. Therefore, deploying Re-ECN in a flow-level or a user-level is practical in LTE.

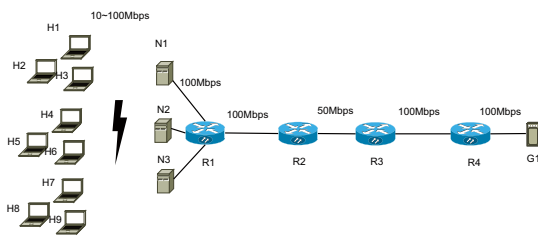## IV. SIMULATION METHODOLOGY

Fig. 4.   Simulation Setup.

We conduct our simulation using NS-2 simulator 2.30 [13] with two additional modules: the UMTS/HSDPA Extension EURANE [14] and the Re-ECN module. There has been a preliminary implementation of Re-ECN based on the "FullTCP" module. We extend this module with the policer and dropper

algorithms and integrate these functionalities into the HSDPA module. In the future, we will improve the simulator to cope with the radio segmentation issue.

Figure 4 depicts our simulation scenario including a number of wireless clients connecting to the network. Every 5 clients are connected to eNodeB with a wireless interface at 100Mbps bandwidth with configurable error rate. eNodeBs are in the access sites connected to the GW using 2 routers along the path. The bandwidth between routers and hosts are 100Mbps except for the link between router R2 and R3. The bottleneck link between router R2 and R3 is 50Mbps. The propagation delays are set to be 200ms on the wireless links and 20ms on the wired links. To study the protocol's sensitivity to latency, we experiment with a variety of propagation delays for this link, ranging from 5 to 200 ms. The TCP session is established between the clients and the PDN-GW directly, traversing the eNodeB and Serving-GW as the overlay nodes. All routers in our simulations use RED with ECN marking. With RED-ECN, the packet drop probability increases linearly from p_max to 1 as the queue size grows. For each experiment, we run the simulation 5 times and take the average of them.

Our evaluation contains both the client-based metrics and the network-based metrics. The client-based metrics include the metric *Goodput*, the total useful data received at the application level by all receivers divided by the simulation time, and the *Response Time or RTT*, the time between when the client's request sent and when the last requested page from a server arrives. The network-wide metrics include *average network utilization*, the ratio of the sum of application payloads across all n flows divided by the amount of data transferred on the link with capacity B during time T: $G = \frac{\sum_{i=0}^{n-1} f_i}{BT}$; and *flow fairness*: Jains fairness index [15], the difference between goodputs among flows as the metric of fairness: $FI(g_0, \ldots g_{n-1}) = \frac{\left(\sum_{i=0}^{n-1} g_i\right)^2}{n \sum_{i=0}^{n-1} g_i^2}$.
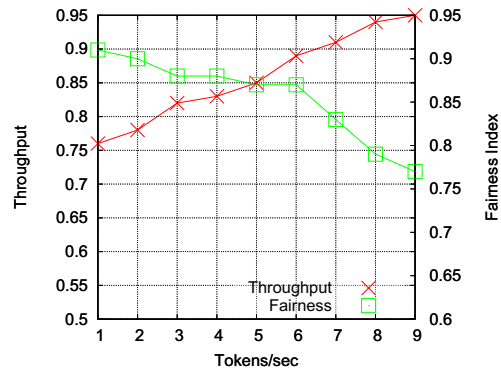

Fig. 5.   Different tokens/sec.

V. SIMULATION BASED RE-ECN ANALYSIS

In this section, we comprehensively evaluate the Re-ECN's performance under different parameter settings and network conditions.
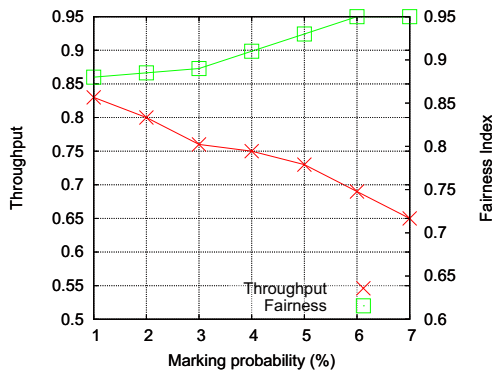
Proceedings of the 2011 23rd International Teletraffic Congress (ITC 2011)
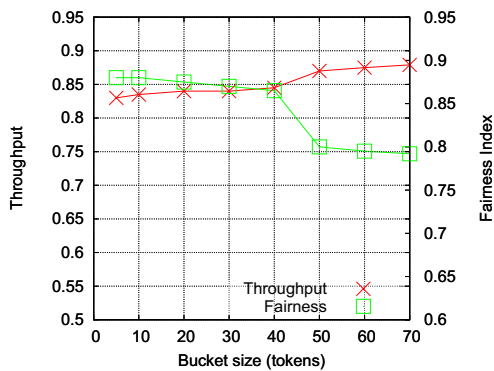
Fig. 6. Different marking probability.

Fig. 7. Different start bucket size.

A. Impact of Re-ECN Parameter Settings

As explained in Section II, to ensure the correct functioning of Re-ECN, the operator needs to configure a few critical parameters in the ECN Marker, Policer, and Dropper. The parameters on Policer includes the token bucket rate r, the bucket initial size b_{start} and the maximum bucket size b_{max}. RED-based ECN Marker has three key parameters on when/how to mark packets, *i.e.,* the marking probability p, the minimum queue size q_{min} and the maximum queue size q_{max} before dropping all packets from the tail. We study the impact of the three Re-ECN related parameters on performance as we focus on evaluating the Re-ECN in this paper. We simulate both 10 and 100 flows and use the total goodput and the fairness index as the evaluation metrics. Our goal is to study how the overall network performance changes with the configuration of various settings. Given the number of parameters to test is large, for each experiment we examine one parameter while fixing the remaining. We will leave the study on the combination of parameters for the future work.

Figure 5 evaluates the token filling rate r. It shows a positive relationship with the throughput: more tokens given by the Policer meaning that more resources are allocated to the flow/user. Figure 7 shows that the goodput increases as b_{start} while the fairness appears the opposite trend. This

is because with larger b_{start}, each flow has more freedom in throughput fluctuation. A larger b_{start} also means less restriction in flow control, leading to worse fairness. In face of greedy flows, higher b_{start} takes longer to identify and punish these flows. Similarly, ECN Marker's marking probability p can also control the restrictiveness. Figure 6 shows that the goodput decreases significantly with the increasing p, showing higher level of control in the network, which ensures a higher degree of fairness.
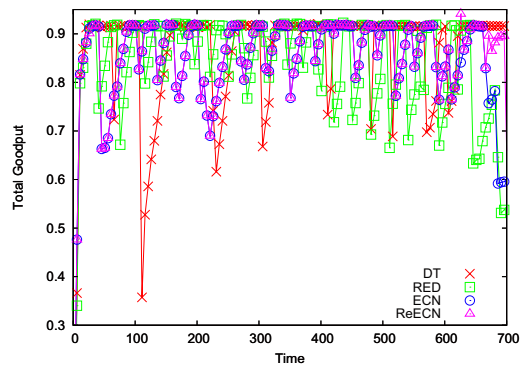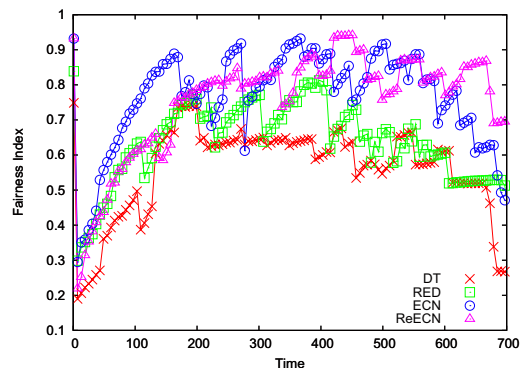

Fig. 8. Throughput comparison.

Fig. 9. Fairness comparison.

We next compare Re-ECN framework with other resource management mechanisms: drop tail based queueing, random early drop based queuing, and the ECN. Figure 8 shows the total throughput relative to the relevant bandwidth utilization compared to the total capacity. The DropTail (DT) shows the largest fluctuation and overall lowest utilizations. RED improves upon DT significantly but is still worse than ECN and Re-ECN where the end-host actively avoids congestion based on ECN marking. The difference between ECN and Re-ECN is not significant, with ECN performs better at the beginning while Re-ECN is better towards the end. The difference is likely due to the TCP slow start process encountering slightly worse performance with Re-ECN. Another main benefit from Re-ECN is to ensure fairness. We first evaluate the fairness using fairness index shown in Figure 9. The benefits of ECN
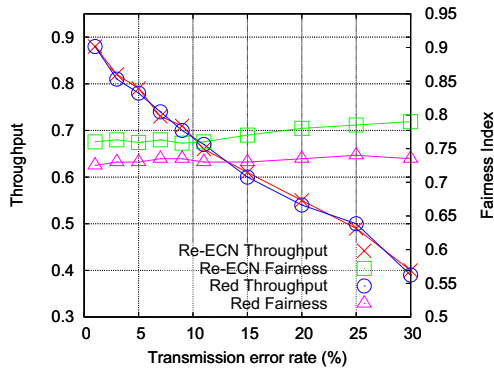
Fig. 10.   Different transmission error rate (Infrastructure-model).

and Re-ECN are significant. DT performs worse as expected. Our studies indicate strongly the improvements in goodput and especially in fairness without significant tradeoffs as we move from DropTail and RED to ECN and Re-ECN.

B. Deployment in LTE scenarios and applications

The previous analysis are simulated using end-to-end model in scenario(a). Next, we study its performance in infrastructure model in scenario(b), *i.e.,* Re-ECN is deployed between the UE and the PDN-GW. Again we show the goodput and the fairness with different transmission error rate. First, we observe that with small error rate, infrastructure based Re-ECN can achieve both high relative throughput as well as high fairness index. Interestingly we found that the benefit of Re-ECN becomes less significant as the transmission error rate on the wireless link increases. Extremely, if the loss/error rate on the air interface is larger than 5%, then the goodput for Re-ECN, ECN and Reno becomes undifferentiated. Figure 10 shows the throughput decreases significantly while the fairness remains nearly the same. This is because of the packet loss induced by transmission error is not an indication of congestion here. However, both ECN and Re-ECN still treat it as a signal for congestion to reduce the sending rate, which results in low network utilization.

In Re-ECN, truthful downstream path information is visible to ingress network operators in data packets, which can respond to incipient congestion in time. This is equivalent to offering different levels of QoS, *e.g.,* premium service with zero congestion response. Similarly, it can also be used for defending against Denial-of-Service (DoS) attacks. We first demonstrate its application of defending against malicious flows with x10 bandwidth consumption. We compare the scenario of ECN enabled end hosts and RED enabled routers, with the scenario of Re-ECN framework, including Re-ECN enabled eNodeBs/GWs, ECN Marker, Policer and Dropper. Figure 11 shows the throughput comparison between attack flows and legitimate victim flows and Figure 12 shows the response time. It shows that the Re-ECN is very efficient in throttling the attack flow and reserving bandwidth for the legitimate victim flows, with larger goodput and smaller

response time. With the Re-ECN being deployed, the victim flows' throughput remains stable in face of an attack.

Another application demonstrated is to provide differentiated services. Figure 13 shows the throughput changes over time for four classes of traffic. We set up four different ECN Marker's marking probability p: $p = \{0.01, 0.05, 0.1, 0.2\}$, mapping to the corresponding class 1-4 from low to high probability. The simulation shows the clear separation and performance difference among different classes measured in throughput.
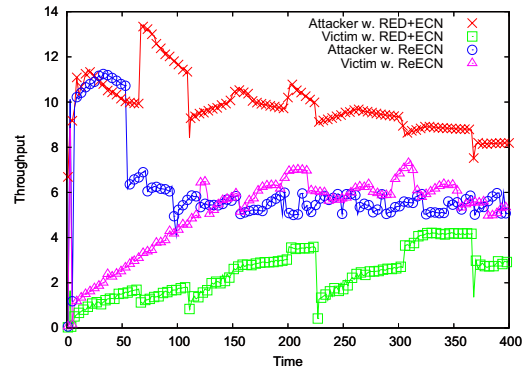

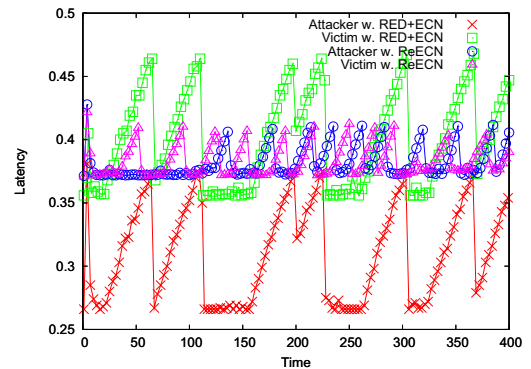Fig. 11.   Throughput with DoS attack.

Fig. 12.   RTT with DoS attack.

VI. RELATED WORK

In its 35 year history, TCP has been repeatedly challenged to adapt to new environments. Researchers have proved adroit in doing so, enabling TCP to function well in gigabit networks [16], long/fat networks [17], [18], and satellite and wireless environments [19], [20]. To improve the performance of TCP in long-distance networks, a number of new TCP variants, including High Speed TCP [21], Scalable TCP [22], and FAST [23], to mention a few, have been proposed. Slim and Ahmed [24] used a Linux-based test bed network to study TCP/ECN. These work are complimentary to ours, the first comprehensive evaluation on Re-ECN.
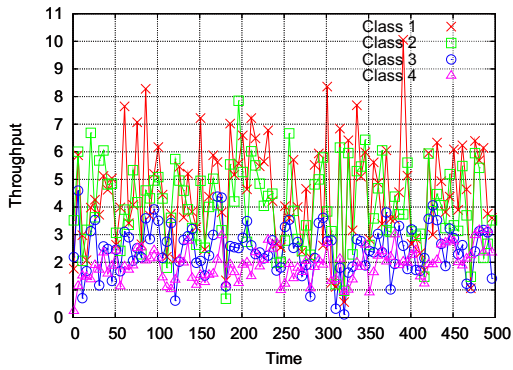
Fig. 13.   Throughput with QoS.

VII. Conclusion

In this paper we propose two different architecture framework of Re-ECN on the LTE networks: end-to-end based and infrastructure based design. We are the first to propose a novel deployment strategy without changes to the end users. We then evaluate the Re-ECN's performance with different network topology, network conditions, and different parameter settings using simulation. We find that Re-ECN has significant advantage under certain network conditions but not much benefit under severe packet loss rate.

References

[1] R. Kalden and H. Ekström, "Searching for mobile mice and elephants in gprs networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 8, no. 4, pp. 37–46, 2004.

[2] A. Botta, A. Pescape, G. Ventre, E. W. Biersack, and S. Rugel, "Performance footprints of heavy-users in 3g networks via empirical measurement," in *WiNMee 2010, 6th International workshop on Wireless Network Measurements, May 31st, 2010, Avignon, France*, 05 2010.

[3] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," in *SIGCOMM '89: Symposium proceedings on Communications architectures & protocols*.   New York, NY, USA: ACM, 1989, pp. 1–12.

[4] M. Allman, V. Paxson, and W. Stevens, "Tcp congestion control," RFC 2581, 1999.

[5] M. Handley, S. Floyd, J. Padhye, and J. Widmer, "Tcp congestion control," RFC 3448, 1999.

[6] B. Briscoe, "Internet: Fairer is faster," BT White Paper TR-CXR9-2009-001, 2009.

[7] A. Jacquet, B. Briscoe, and T. Moncaster, "Policing Freedom to Use the Internet Resource Pool," in *Proc Workshop on Re-Architecting the Internet (ReArch'08)*.   ACM, December 2008, (to appear).

[8] B. Briscoe, A. Jacquet, T. Moncaster, and A. Smith, "Re-ECN: Adding accountability for causing congestion to TCP/IP," Internet Engineering Task Force, Tech. Rep. draft-briscoe-tsvwg-re-ecn-tcp-08.txt, September 2009, (Work in progress).

[9] ——, "Re-ECN: The motivation for adding congestion accountability to TCP/IP," Internet Engineering Task Force, Internet Draft draft-briscoe-tsvwg-re-ecn-tcp-motivation-01.txt, September 2009, (Work in progress).

[10] K. Ramakrishnan, S. Floyd, and D. Black, "The addition of explicit congestion notification (ecn) to ip," RFC 3168, 2001.

[11] M. Bagnulo and N. Dukkipati, "Congestion exposure (conex) working group," https://datatracker.ietf.org/wg/conex/charter/.

[12] B. Briscoe, "Tunnelling of explicit congestion notification," https://datatracker.ietf.org/wg/conex/charter/.

[13] "The network simulator ns-2," http://www.isi.edu/nsnam/ns.

[14] "Eurane," http://www.ti-wmc.nl/eurane.

[15] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," DEC Research, Tech. Rep. TR-301, September 1984.

[16] C. Partridge, *Gigabit networking*.   Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1994.

[17] L. S. Brakmo, S. W. O'Malley, and L. L. Peterson, "Tcp vegas: new techniques for congestion detection and avoidance," *SIGCOMM Comput. Commun. Rev.*, vol. 24, no. 4, pp. 24–35, 1994.

[18] V. Jacobson, R. Braden, and D. Borman, "Tcp extensions for high performance," United States, 1992.

[19] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. H. Katz, "A comparison of mechanisms for improving tcp performance over wireless links," in *SIGCOMM '96: Conference proceedings on Applications, technologies, architectures, and protocols for computer communications*.   New York, NY, USA: ACM, 1996, pp. 256–269.

[20] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, "Tcp selective acknowledgement options," United States, 1996.

[21] S. Floyd, "Highspeed tcp for large congestion windows," United States, 2003.

[22] T. Kelly, "Scalable tcp: improving performance in highspeed wide area networks," *SIGCOMM Comput. Commun. Rev.*, vol. 33, no. 2, pp. 83–91, 2003.

[23] D. X. Wei, C. Jin, S. H. Low, and S. Hegde, "Fast tcp: motivation, architecture, algorithms, performance," *IEEE/ACM Trans. Netw.*, vol. 14, no. 6, pp. 1246–1259, 2006.

[24] Y. Zhang and L. Qiu, "Understanding the end-to-end performance impact of red in a heterogeneous environment," Ithaca, NY, USA, Tech. Rep., 2000.