Observations of Bufferbloat in Swedish Cellular Networks

Stefan Alfredsson[1], Giacomo Del Giudice[2], Johan Garcia[1], Anna Brunstrom[1], Luca De Cicco[2,3], Saverio Mascolo[2]
[1]Karlstad University, Sweden, [2]Politecnico di Bari, Italy, [3]Ecole superieure d'Electricite, France
{Stefan.Alfredsson, Johan.Garcia, Anna.Brunstrom}@kau.se, dggiacomo@gmail.com, ldecicco@gmail.com, mascolo@poliba.it

Abstract—**The existence of excessively large and too filled network buffers, known as bufferbloat, has recently gained attention as a major performance problem for delay-sensitive applications. One important network scenario where bufferbloat may occur is cellular networks. This paper investigates the interaction between TCP congestion control and buffering in cellular networks. Extensive measurements have been performed in a commercial 3.5G (HSPA+) cellular network, with a mix of long and short TCP flows using the CUBIC, NewReno and Westwood+ congestion control algorithms. The results show that the completion times of short flows increase significantly when concurrent long flow traffic is introduced. This is caused by increased buffer occupancy from the long flows. The completion times are shown to depend significantly on the congestion control algorithms used for the background flows, with CUBIC leading to significantly larger completion times compared to TCP Westwood+.**

I. INTRODUCTION

Long queues and additional buffering in the network can be used to increase link utilization and reduce download times. Recently there has, however, been a growing awareness within the networking community that too much buffering may cause problems for delay-sensitive applications. Excessively large and often full buffers, referred to as "bufferbloat", is now recognized as a serious problem in the Internet [1]. Widespread severe over-buffering has also been reported for several parts of the Internet [2], [3], [4], [5].

Bufferbloat results in significantly reduced responsiveness of applications because of excess buffering of packets within the network. It causes both high latency and can also result in appreciable jitter [6]. This is particularly problematic for short TCP flows such as Web traffic or real-time interactive UDP traffic such as VoIP. When such traffic shares resources with greedy TCP transfers it ends up at the end of a full transmission buffer and experiences an increased delay that can severely deteriorate user performance [7].

Cellular networks are becoming an increasingly important Internet access technology. To accommodate varying data rates over time-varying wireless channels they are also normally provisioned with large buffers [8], [9]. The fact that cellular networks typically employ individual buffer space for each user [9], [10] in combination with a low level of user multitasking over cellular connections has in the past limited the impact of these buffers on user performance. However, with the emergence of more and more powerful smartphones, as well as the increasing use of cellular broadband connections for residential Internet access, multitasking over cellular connections is today becoming common. This makes bufferbloat in cellular networks an increasingly important problem. The recent study by Jiang et. al. [5] also confirm that bufferbloat can lead to round trip times (RTTs) in the order of seconds for cellular networks.

The extent of buffer buildup is determined by the rate of incoming packets versus the rate of outgoing packets. Standard TCP congestion control probes the available bandwidth by injecting packets into the network until there is packet loss, which for tail-drop queuing happens when buffers are full. The way buffers fill up are thus highly dependent on the transport protocol behavior and varies between different TCP congestion control algorithms. To further the understanding in this area, we in this paper investigate the interaction between congestion control and buffering in cellular networks.

II. MEASUREMENT SETUP

To examine the impact of the congestion control on bufferbloat in cellular networks, we have carried out a number of measurements on short flows, long flows, and combinations thereof. These are aimed to represent a usage where multiple flows of different lengths can be active simultaneously, as can be expected in use cases related to cellular-based residential Internet access, and when smartphones simultaneously access both background data and the Web or other interactive services.

A. Metrics

Web browsing is an important application in many use cases. Web page response time is thus a key factor, and in this paper it is used as a primary performance metric. The Web page size obviously affects the response time. In [11] the average data transfer per Web page is reported to be 320 kB, which is typically split into several different host connections. For short transfers, the experiments in this paper use 320 kB of data, but sent in a single flow. For a real-world scenario, this could represent downloading the base HTML-page of a popular news site such as Dagens Nyheter[1]. In this way, it is possible to better study the interaction between a short-lived Web flow and a long-lived background flow. In addition, metrics such as goodput, packet loss ratio (PLR), and round-trip time (RTT) are also reported when appropriate.

B. Measurement collection

The experimental campaign was carried out over three different cellular network technologies (3G, 3.5G and 4G) all

---

[1] As of 5 april 2013, www.dn.se was 319145 bytes.

provided by the same operator in Sweden. In this paper, only the results for 3.5G is reported, while [?] shows the results also for 3G and 4G. To perform the experiments two computers were used, one laptop with a Huawei E392 USB modem and one server with a connection to the Swedish University backbone. Both computers were running Ubuntu 12.04 with kernel versions 3.1.4 on the laptop and 3.2.0-27 on the server.

A set of measurements were performed to collect data both on the network characteristics for single flows of varying lengths, as well as when short-lived flows are competing with long-lived background flows. Data was collected using tcpdump and later processed with tcptrace to extract flow related information and the relevant metrics.

C. *Measurement configuration*

A wide range of measurements were made as illustrated in Figure 1. Runs A and B collect baseline information by measuring transfer characteristics for long and short-lived flows without any competing traffic. Run A uses long-lived flows to collect data for 3 minutes for each of the three considered congestion control algorithms, and with three replications each. Similarly, run B collects data for the 320 kbyte short flows, but with 10 replications. In contrast to runs A and B, run C involves both a long-lived background flow and short-lived flows. Figure 1 shows that all combinations of congestion controls for the short and long flows are evaluated, again with 10 replications for each short flow congestion control. Finally, run D differs from run C in that five concurrent long flows are now used to generate the background traffic instead of a single flow. Thirteen measurement runs were concatenated into a single measurement campaign with the following run composition: ABCDABCDABCDA, further increasing the number of replications, and interleaving measurements to reduce time-of-day effects.

III. RESULTS

In this section the results from our 3.5G measurements are presented.

A. *Overview of results*

Figure 2 shows an overview of the results from measurement runs B, C, and D. The figure shows the average Web response time and the 95% confidence interval of 30 repeated short flows for each configuration. Shorter time is better. Leftmost in the figure, the first three bars show the average Web response times without concurrent traffic. Second, the middle group of nine bars shows the Web response times of the short flows, with one background flow, and all combinations of the three congestion control algorithms. Third, the right group of nine bars also shows the Web response times but for five background flows, for all combinations of congestion controls.

With no concurrent flows the average Web response times are around 0.9 seconds for all three congestion controls. When one background flow is introduced (shown in the second group), the Web response times increase five-fold when CUBIC or NewReno is used for the background flow. When Westwood+ is used, the increase in Web response times is smaller, with roughly a doubling of the response times. The choice of congestion control for the short flow is, however, of lesser importance, compared to the congestion control for the background flow.

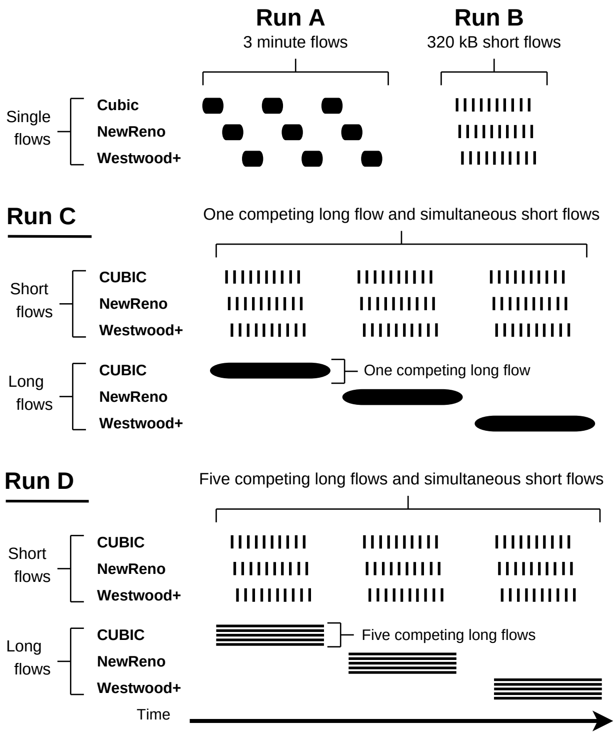


Fig. 1. Measurement setup

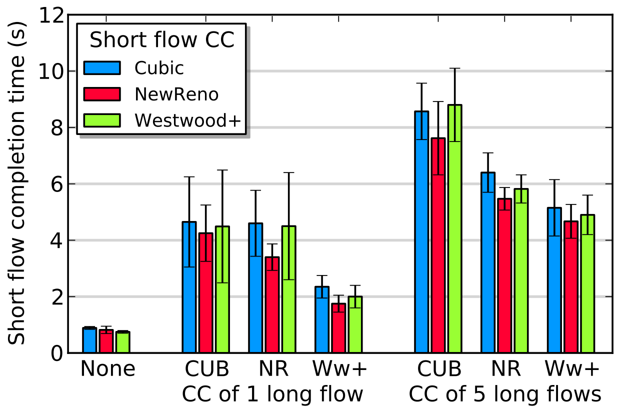
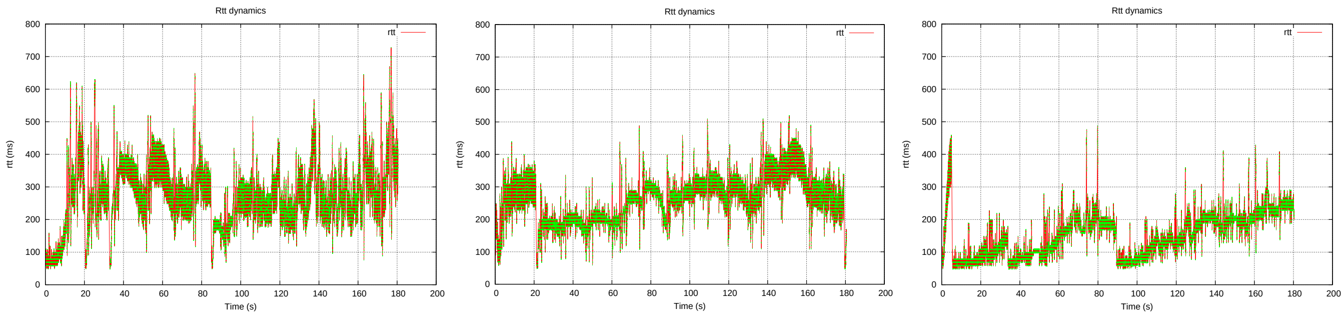


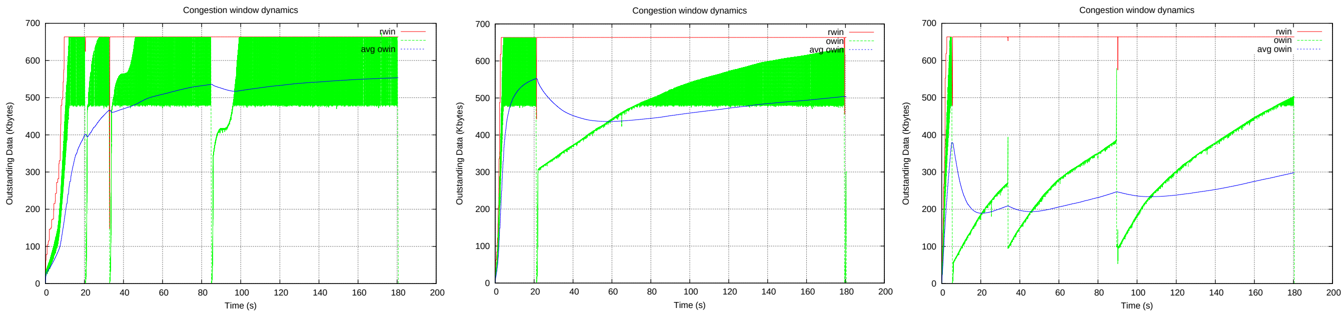
Fig. 2. The average Web response time of short flows using CUBIC, NewReno and Westwood+ over 3.5G.

When five background flows are introduced, this increases the Web response times of the short flows even further; to about 8 seconds for CUBIC, about 6 seconds for NewReno and about 5 seconds for Westwood+. The same trend exists where the congestion control of the short flow is of lesser importance, compared to the congestion control of the long



(a) CUBIC (b) NewReno (c) Westwood+

Fig. 4. The round-trip time of single long CUBIC, NewReno and Westwood+ flows, over 3.5G.



(a) CUBIC (b) NewReno (c) Westwood+

Fig. 5. The amount of outstanding data of single long CUBIC, NewReno and Westwood+ flows, over 3.5G.

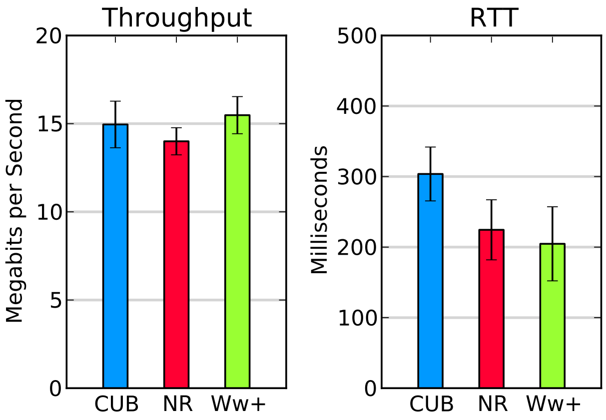


Fig. 3. The average throughput and round-trip times of single long flows that use CUBIC, NewReno and Westwood+, over 3.5G, without concurrent traffic.

flow.

From these graphs we observe that our measurements are indeed prone to bufferbloat. The introduction of more background flows has a severe impact on the short flows and results in higher Web response times. Additionally, the congestion control algorithm used for the background flow also clearly affects the Web response time, where a CUBIC background flow results in significantly higher Web response times compared to a Westwood+ background flow. Varying the congestion control for the short flow does, however, not result in any significant differences in response time.

B. Analysis

To further analyze the underlying cause for the difference in Web response times we examine the characteristics of long flows for the different congestion control algorithms. Figure 3 shows the goodput and the round-trip times of only long flows (Run A). As shown in the figure, there is no significant difference in goodput between the three congestion control algorithms. However the corresponding round-trip times in the figure differ considerably: the average round-trip time for CUBIC long flows in this scenario is 1.5 times longer than that of Westwood+ long flows, with NewReno in-between.

To illustrate this in more detail, Figures 4(a) – 4(c) show the round-trip time dynamics for the duration of a CUBIC, NewReno and Westwood+ flow, respectively. While there is a lot of variation in the round-trip times, the difference in round-trip times between different congestion controls is also visible here. The explanation can be found in the graphs showing the outstanding windows for the respective flows, Figure 5(a) – 5(c). Comparing these, the average outstanding window of the CUBIC flow is roughly twice as large as for the Westwood+ flow after the TCP startup phase, with NewReno in-between.

The throughput of the flows studied in Figures 4 and 5 is measured at 16.1 Mbit/s for the CUBIC flow, FIXME Mbit/s for the NewReno and 15.7 Mbit/s for the Westwood+ flow, i.e. the throughput is almost the same. This means that the available capacity is utilized in a similar way, and that the additional outstanding data in the case of CUBIC is stored in buffers in the network. Put in another way, increasing the amount of outstanding data does not increase the throughput (if the path capacity is reached), but fills buffers, increases the queueing delay, and leads to the observed increase in round-trip times.

Even though there is little difference in throughput, the buffered outstanding data and the increased round-trip times in the case of CUBIC cause performance degradation when concurrent flows are introduced. Figure 6 shows the round-trip times of short CUBIC flows, corresponding to the Web response times shown in the overview in Figure 2. Without any background flow, the average round-trip time of a CUBIC short flow is about 90 ms. When a background flow is present, the round-trip time is increased, in the same magnitude as the Web response time. The way data is buffered in the network by the background flow thus affects the round-trip time and, as a consequence, the response time of concurrent short flows.

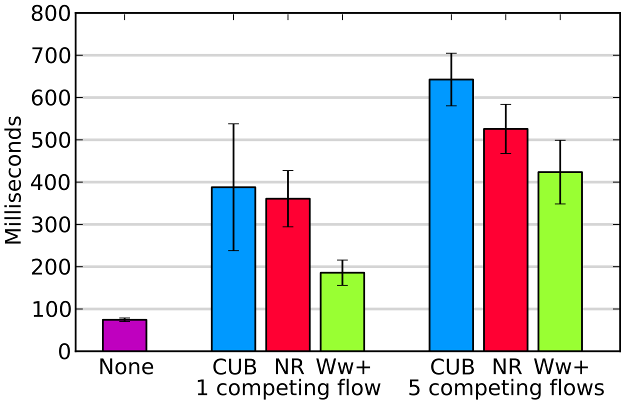


Fig. 6. The average round-trip times of short flows using CUBIC with 0, 1 and 5 background flows that use CUBIC, NewReno and Westwood+, over 3.5G.

IV. CONCLUSIONS

The rise in cellular-based residential Internet access as well as the increased usage and evolved applications of smart phones are driving demand for cellular Internet connectivity and capacity. In these use cases applications commonly use background data transfer, or longer running non-interactive transfers are mixed with shorter user-interactive transfers. In this context the amount of buffering in the network becomes relevant, and especially with regards to bufferbloat, i.e. excessively large and filled buffers.

We examined the presence and impact of bufferbloat in cellular networks by performing a large-scale measurement campaign using the cellular networks of a major Swedish provider, focusing here on the results from the 3.5G network. The results showed that the impact of the used congestion control, i.e. NewReno, Westwood+ and CUBIC, was not a major factor when there was only a single flow utilizing the link. In contrast, when shorter Web flows are mixed with longer running background flows the congestion control of the background flows was shown to have a major impact in the measured environment. If a background flow uses Westwood+ the penalty to a competing short Web flow is roughly halved as compared to CUBIC. These results would seem to suggest that in order to optimize the user perceived responsiveness, servers providing background data to cellular devices should use Westwood+ or other less aggressive congestion control algorithms while Web servers and similar may use CUBIC as congestion control. However, before providing any recommendations more measurements using a wider range of operators and networks are necessary. For future work we plan to perform measurement campaigns in additional networks as well as examine other congestion controls such as LEDBAT [12].

V. ACKNOWLEDGMENT

This work was partly funded by the European Community under its Seventh Framework Programme through the Reducing Internet Transport Latency (RITE) project (ICT-317700) and partly funded by .SE Internetfonden. The views expressed are solely those of the authors.

REFERENCES

[1] "Bufferbloat Home Page," URL: http://www.bufferbloat.net/ (Apr 2013).
[2] M. Dischinger, A. Haeberlen, K. Gummadi, and S. Saroiu, "Characterizing Residential Broadband Networks," in *Proceedings of ACM IMC*, 2007.
[3] C. Kreibich, N. Weaver, B. Nechaev, and V. Paxson, "Netalyzr: Illuminating the Edge Network," in *Proceedings of ACM IMC*, 2010.
[4] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè, "Broadband Internet Performance: A View From the Gateway," in *Proceedings of ACM SIGCOMM*, 2011.
[5] H. Jiang, Y. Wang, K. Lee, and I. Rhee, "Tackling Bufferbloat in 3G/4G Networks," in *Proceedings of ACM IMC*, 2012.
[6] J. Gettys and K. Nichols, "Bufferbloat: Dark Buffers in the Internet," *Communications of the ACM*, vol. 55, no. 1, 2012.
[7] I. Järvinen, B. Chemmagate, L. Daniel, A. Y. Ding, M. Kojo, and M. Isomäki, "Impact of TCP on Interactive Real-Time Communication," IAB/IRTF Workshop on Congestion Control for Interactive Real-Time Communication, July 2012.
[8] R. Chakravorty, J. Cartwright, and I. Pratt, "Practical experience with TCP over GPRS," in *Proceedings of IEEE GLOBECOM*, Taiwan, Nov. 2002.
[9] X. Liu, A. Sridharan, S. Machiraju, M. Seshadri, and H. Zang, "Experiences in a 3G Network: Interplay Between the Wireless Channel and Applications," in *Proceedings of ACM MOBICOM*, USA, 2008.
[10] M. Sågfors, R. Ludwig, M. Meyer, and J. Peisa, "Queue management for TCP traffic over 3G links," *Proceedings of IEEE WCNC*, 2003.
[11] S. Ramachandran, "Web metrics: Size and number of resources," https://developers.google.com/speed/articles/web-metrics, May 2010.
[12] S. Shalunov, G. Hazel, J. Iyengar, and M. Kuehlewind, "Low Extra Delay Background Transport (LEDBAT)," RFC 6817 (Experimental), Internet Engineering Task Force, Dec. 2012.