On the Use of TCP Passive Measurements for Anomaly Detection: A Case Study from an Operational 3G Network

Peter Romirer-Maierhofer[1], Angelo Coluccia[2], and Tobias Witek[1]

[1] Forschungszentrum Telekommunikation Wien (FTW), Austria.
[2] Università del Salento, Italy.
email: lastname@ftw.at

Abstract. In this work we discuss the use of passive measurements of TCP performance indicators in support of network operation and troubleshooting, presenting a case-study from a real 3G cellular network. From the analysis of TCP handshaking packets measured in the core network we infer Round-Trip-Times (RTT) on both the client and server sides separately for UMTS/HSPA and GPRS/EDGE sections. We also keep track of the relative share of packet pairs which did not lead to a valid RTT sample, e.g. due to loss and/or retransmission events, and use this metric as an additional performance signal. In a previous work we identified the risk of measurement bias due to early retransmission of TCP SYNACK packets by some popular servers. In order to mitigate this problem we introduce here a novel algorithm for dynamic classification and filtering of early retransmitters. We present a few illustrative cases of abrupt-change observed in the real network, based on which we derive some lessons learned about using such data for detecting anomalies in a real network. Thanks to such measurements we were able to discover a hidden congestion bottleneck in the network under study.

1 Motivations

The evolving nature and functional complexity of a 3G network increases its vulnerability to network problems and errors. Hence, the timely detection and reporting of network anomalies is highly desirable for operators of such networks. Passive packet-level monitoring can be an important instrument for supporting the operation and troubleshooting of 3G networks. A natural approach to validating the health status of a network is the extraction of performance indicators from passive probes, e.g. Round-Trip Time (RTT) percentiles and/or frequency of retransmission. These indicators, which we hereafter refer to as "network signals", can be analyzed in real time in order to detect anomalous deviations from the "normal" network performance observed in the past. This approach underlies two fundamental assumptions: *i*) that extracted network signals are stable over time under problem-free operation, and *ii*) that anomalous phenomena generate appreciable deviations in any of the observed signals. In an earlier work [3] we demonstrated that passively extracted TCP RTT distributions are relatively

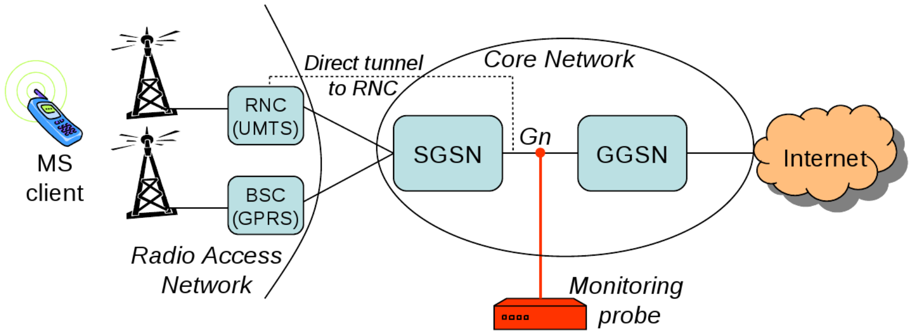
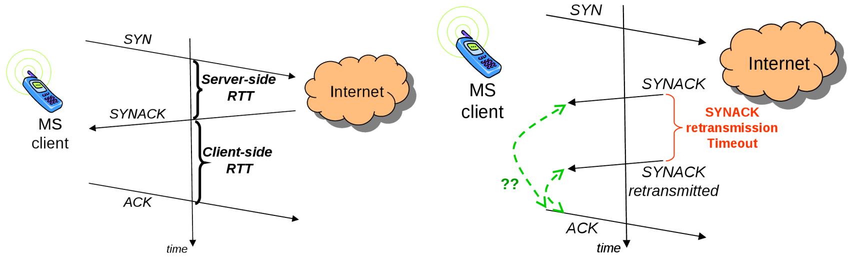


Fig. 1. Monitoring setting.

stable over time in the operational network under study. Here, we take the next step and present some cases from a real 3G network where abnormal events were reflected by a sudden change in the analyzed network signals. Our findings are promising about the possibility of leveraging passively extracted TCP performance indicators for troubleshooting of real 3G networks.

The TCP performance indicators presented in this work are obtained from the passive analysis of TCP handshaking packets. The idea of measuring TCP performance by observing handshaking packets was already presented in 2004 by Benko *et al.* [1] who reported results from an operational GPRS network. Vacirca *et al.* [2] reported RTT measurements from an operational 3G network including GPRS and also UMTS. In [3] we have shown that RTT values have decreased considerably due to the introduction of EDGE and HSPA and the consequential increase of radio bandwidth. Several studies [4–7] presented passive estimation of TCP RTT in wired networks inferred also from TCP DATA/ACK pairs. However, this approach is complicated by loss, reordering and duplication of TCP segments as well as by delayed acknowledgements [4]. Jaiswal *et al.* [5] measure TCP RTT by keeping track of the current congestion window of a connection by applying finite state machines (FSM). Since the computation of the congestion window differs among different flavors of TCP, the authors suggest the parallel operation of several FSMs, each tailored to a specific TCP flavor. Rewaskar *et al.* [6] identified Operation System (OS)-specific differences in prominent TCP implementations, which may bias the passive analysis of TCP segment traces if not handled properly. This issue is addressed by implementing four OS-specific state machines to measure TCP RTT while discarding all connections with less than 10 transmitted segments. Mellia *et al.* [7] compute the TCP RTT by applying the moving average estimator standardized in [8]. In case of short TCP flows as e.g. client HTTP requests, no RTT samples may be collected by this approach [8]. As shown in [4], the RTT inferred from TCP handshake packets is a reasonable approximation of the minimum RTT of the whole connection. Motivated by this result, we elaborate on the use of such RTT measurements for long-term and real-time anomaly detection in an operational 3G network.



(a) Computation of RTT (b) Ambiguity by retransmission

Fig. 2. Measurement schemes.

We believe this method to be much more scalable, since it neither requires the analysis of all packets of a TCP flow nor it relies on any knowledge about the involved TCP flavors and/or Operating Systems. Moreover, in contrast to [6, 7], this approach does not exclude short TCP flows from RTT measurements.

The detection of congestion bottlenecks by passively inferring spurious retransmission timeouts from DATA/ACK pairs was presented in [9]. In this work we show that our simpler approach of extracting RTT just from TCP handshake packets is also suitable to detect hidden congestion bottlenecks.

2 Measurement Methodology

The measurement setting is depicted in Fig. 1. Packet-level traces are captured on the so-called “Gn interface” links between the GGSN and SGSN — for more information about the 3G network structure refer to [10]. We use the METAWIN monitoring system developed in a previous research project and deployed in the network of a mobile operator in EU — for more details refer to [11]. By extracting and correlating information from the 3GPP layers (GTP protol on Gn, see [12]) the METAWIN system enables discrimination of connections originated in the GPRS/EDGE and UMTS/HSPA radio sections.

We shortly recap the RTT measurement methodology already introduced in [3]. We only consider TCP connections established in uplink, i.e. initiated by Mobile Stations in the Radio Access Network. By measuring the time span between the arrival of a SYN and the arrival of the associated SYNACK we infer the (semi-)RTT between the Gn link and a remote server in the Internet — denoted by “server-side RTT” in Fig. 2(a). Similarly, we estimate the (semi-)RTT in the Radio Access Network (RAN), between the Gn link and the Mobile Station — referred to as “client-side RTT”— by calculating the time span between the arrival of the SYNACK and the associated ACK. Valid RTT samples may only be estimated from unambiguous and correctly conducted 3-way handshakes. Those cases where the association between packet pairs is ambiguous (e.g. due to retransmission, duplication) have to be discarded. Within a measurement interval (e.g. 5 minutes) valid RTT samples are aggregated into equal-sized bins.

The corresponding bin width is 0.1 ms for RTT samples <100 ms and 1 ms for RTT samples $\geq$ 100 ms. This two-level binning keeps the total number of bins reasonably low while offering sufficiently accurate resolution for both, lower RTT samples typically measured at the server-side and higher RTT samples typically measured at the client-side. We collect RTT samples separately for TCP handshakes to port 80 and to all other ports. As shown in § 3.4, this differentiation is motivated by the fact that part of the traffic to port 80 might be intercepted by a network-wide proxy.

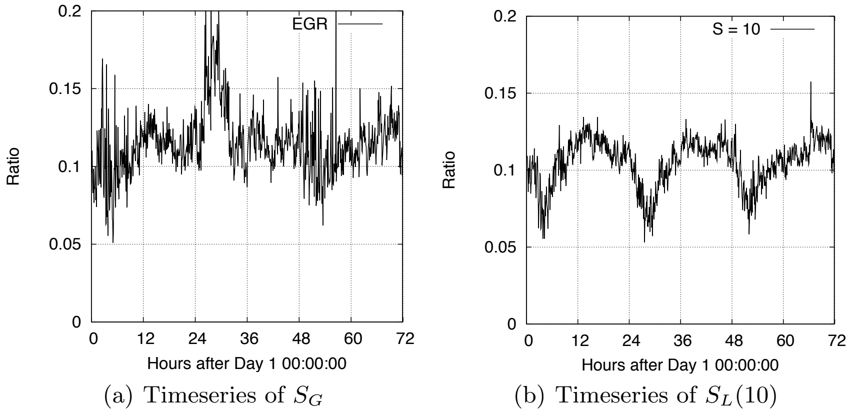
2.1 Invalid Sample Ratio

We mentioned above the problem of ambiguous association of handshake pairs, which can be caused by different reasons. One example of retransmissions is depicted in Fig. 2(b). When the SYNACK retransmission timer expires before an ACK is received, the server retransmits a second SYNACK and this leads to ambiguity in the association SYNACK/ACK: it cannot be decided whether the ACK packet is acknowledging the first SYNACK (correctly received) or the second one (in case the first one was lost along the path). Similar ambiguities can occur for SYN/SYNACK pairs, if e.g. a SYNACK is lost and the client retransmits a second SYN before the expiration of SYNACK retransmission timer at the server. In each time interval (e.g. 5 minutes) we record the relative share of ambiguous SYN/SYNACK and SYNACK/ACK pairs, which we denote by IS_{SYN} and IS_{SYNACK} respectively. Since a retransmission timeout may expire due to loss of the first SYNACK packet in the Radio Access Network (RAN), i.e. on the client-side of the monitored path, the IS_{SYNACK} indicators correlates to — and can be used as proxy signal for — the level of packet loss in the radio section. However, as stated above, ambiguous pairs may be due also to other causes, and the actual level of packet loss will in general stay below the value of IS_{SYNACK}. Similar considerations apply to IS_{SYN} for the server-side section. Focusing on IS_{SYNACK}, we will show that the presence of so called "early retransmitting servers" has a non-negligible influence on such signal. Nonetheless, we expect that an anomalous event raising the network-wide packet loss will be reflected in anomalous deviations of IS_{SYNACK}. In the following, we introduce an indicator built upon IS_{SYNACK} which can be used to reveal anomalous loss events in the network.

In the generic measurement time bin, for each active user i we denote by m_i the number of invalid samples, i.e. SYNACK which could not be univocally associated to an ACK, and by n_i the total number of SYNACK. A simple indicator for IS_{SYNACK} can be defined as the total ratio of invalid samples across all terminals:

$$S_G \overset{\text{def}}{=} \frac{\sum_{i=1}^{I} m_i}{\sum_{i=1}^{I} n_i} \tag{1}$$

where I denotes the total number of active terminals. However, the uneven distribution of n_i — which is typically heavy-tailed — injects a large amount



(a) Timeseries of S_G

(b) Timeseries of $S_L(10)$

Fig. 3. Timeseries of estimation of invalid sample ratio, 3 days, 5 min bins.

of variance into S_G: since most of the terminals have low traffic (low n_i), a few terminals with high traffic (high n_i) and high loss level (high m_i, due e.g. to bad radio conditions, self-congestion or other terminal-specific reasons) might occasionally inflate the value of S_G. This results in a very noisy signal, which complicates the detection of network-wide anomalies. This is clearly visible in the example of Fig. 3(a), which shows a three-day timeseries of S_G. In a previous work [13], we derived a low-variance indicator by taking the weighted average of individual (per terminal) ratios, formally:

$$S_L(\theta) = \sum_{i=1}^{I} \tilde{w}_i \frac{m_i}{n_i} \tag{2}$$

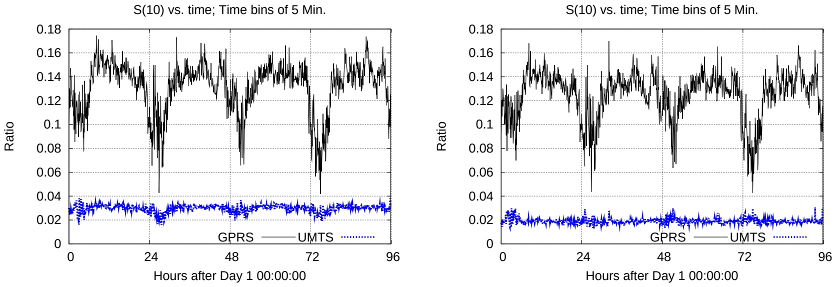
with

$$\tilde{w}_i \overset{\text{def}}{=} \frac{\tilde{n}_i}{\sum_{j=1}^{I} \tilde{n}_j}, \quad \tilde{n}_i \overset{\text{def}}{=} \min(n_i, \theta). \tag{3}$$

The cut-off parameter θ must be set heuristically — it is shown in [13] that such setting is not too much critical. In this work we have chosen $\theta = 10$. In Fig. 3(b) we plot $S_L(10)$ for the same time period as in Fig. 3(a). We observe that $S_L(10)$ — which we call *Invalid Sample Ratio* (ISR) hereafter — provides a much clearer signal than S_G.

2.2 Impact of Early Retransmitted SYNACK Packets

In [3] we identified the risk of a possible measurement bias due to early retransmission of SYNACK. In fact, some popular servers — which we hereafter refer to as *early retransmitters* — resend the SYNACK after 300-500 ms instead of 3 sec which is the recommended value for the TCP Retransmission TimeOut (RTO) [14]. This strategy, which aims at being more responsive against ACK losses, causes an excess of spurious SYNACK retransmissions for wireless



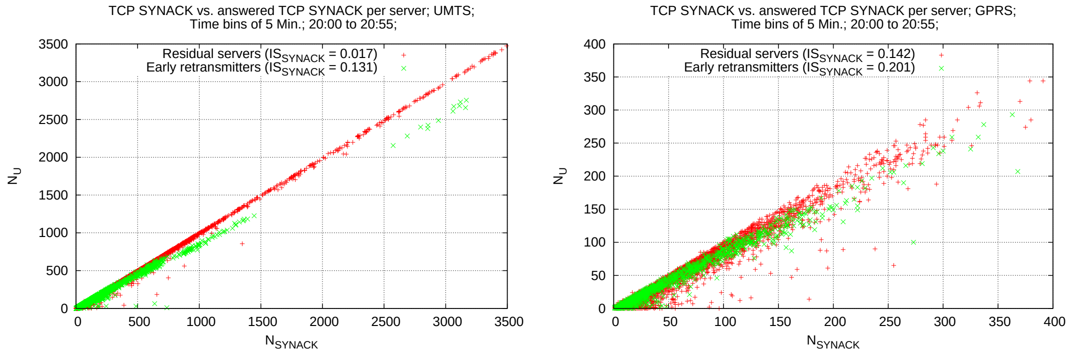
(a) All servers (b) Without early retransmitters

Fig. 4. Invalid sample ratio $S_L(10)$, 4 days, 5 min bins.

connections with high delays such as GPRS/EDGE, and therefore generates a high number of ambiguous SYNACK/ACK associations as the one outlined in Fig. 2(b). Therefore, we can infer the client-side RTT between the SYNACK of an early retransmitter and the associated ACK only if the ACK packets arrive within a retransmission timeout (RTO) of 300-500 ms. All other RTT samples are discarded due to ambiguity of the retransmitted SYNACK packets. This is particularly a problem in GPRS, where 50% of the client-side RTT samples are usually above 500 ms (ref. [3]) and hence, a significant fraction of ACK packets may not arrive in time before the early retransmission of SYNACK. As a consequence, measurements involving early retransmitters include only client-side RTT samples below 300-500 ms, while higher RTT samples are invalidated by retransmissions of the SYNACK.

From the discussion above, it is clear that early retransmitters may bias summary statistics as e.g. client-side RTT percentiles and ISR. Although a shifted indicator with a fixed offset bias might not be a problem for the task of detecting abrupt changes, the biasing effect of early retransmitters is more problematic since the traffic share of early retransmitters can change during time, resulting in artifactual deviations which are not mirroring network anomalies. For that reason, we introduce a simple algorithm for the dynamic classification and filtering of early retransmitters described in the following section aimed at mitigating their influence on the final measurements.

As an example of possible bias introduced by early retransmitters we plot in Fig. 4 two four-day timeseries of ISR (i.e. $S_L(10)$) for traffic to port 80, separately for GPRS/EDGE and UMTS/HSPA. We observe that the ISR for UMTS/HSPA is very stable at a value of around 0.03 before filtering the early retransmitters (ref. Fig. 4(a)) and reduces to 0.02 after filtering (ref. Fig. 4(b)). Hence, the presence of early retransmitters introduces a relative error of 50% of the ISR of UMTS/HSPA. In case of GPRS/EDGE the ISR signal yields a cyclic time-of-day variation between around 0.06 and 0.16. Comparing Fig. 4(a) and 4(b) we observe that early retransmitters introduce an absolute error of 0.01-0.02 in the case of GPRS/EDGE.



(a) UMTS/HSPA

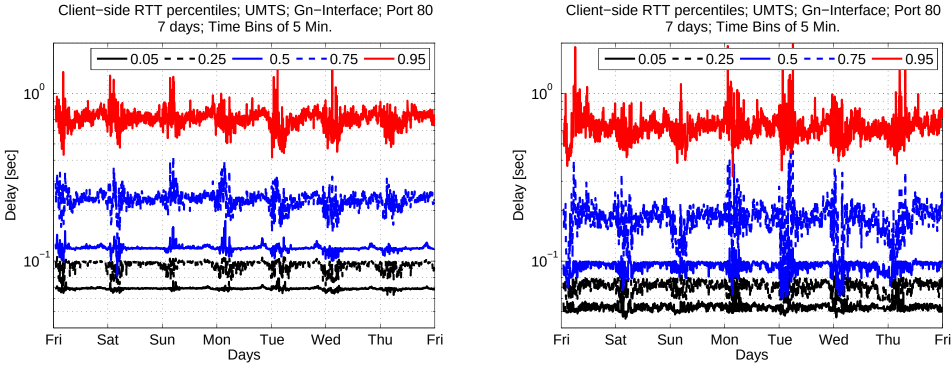
(b) GPRS/EDGE

Fig. 5. Total number of SYNACK vs. number of unambiguously replied SYNACK, Port 80, 5 min bins.

2.3 Filtering of Early Retransmitting Servers

In order to mitigate the statistical bias introduced by early retransmitters, we implement a dynamic classification and filtering of these servers, described in the following. For each measurement interval we count the total number of SYNACK denoted by N_{SYNACK} and the number of SYNACK retransmitted after a retransmission time out <600 ms denoted by N_{EARLY}. We define the early retransmission ratio $r = \frac{N_{EARLY}}{N_{SYNACK}}$. Within an observation period of 1 minute we compute a ratio $r_i(k)$ for each server separately. Finally, a server is classified as early retransmitter if $r_i(k) > 0.01$. If an early retransmitter did not send any SYNACK within the last five observation periods (i.e. 5 minutes), we remove it from the class of early retransmitters. Instead of discarding those measurements which involved early retransmitters, we collect them in a separate class.

Let N_U denote the number of unambiguously replied SYNACKs. Recall that a SYNACK is unambiguously replied only if a client ACK arrives within the SYNACK retransmission timeout of the destined server (ref. Fig. 2(b)). In Fig. 5 we plot for each server N_{SYNACK} versus N_U over a measurement period of one hour in time bins of 5 minutes for handshakes established to port 80. Measurement points of early retransmitters are represented by a green 'x', while residual servers are depicted by a red '+'. We observe two separate clusters in Fig. 5(a) for $N_U > 700$. The points of early retransmitters show a clear offset towards higher values of N_{SYNACK}, since a significant number of SYNACKs is ambiguously replied due to early retransmission of the SYNACK packets. In contrast to that, the points of residual servers are located along the line where each SYNACK is unambiguously replied by an ACK packet (i.e. $N_{SYNACK} \approx N_U$). Interesting to note, both clusters are overlapping for $N_U < 700$. This might be explained by the fact that few early retransmissions per server are already sufficient to exceed our classification threshold of $r_i(k) > 0.01$ if this server is sending a



(a) Path A via SGSN and RNC

(b) Path B via RNC only

Fig. 6. Client-side RTT percentiles in UMTS, different paths, 7 days, 5 min bins.

low number of N_{SYNACK}, which leads to false negatives in our classification method. Moreover, there might be intervals where early retransmitters do not retransmit any SYNACKs, because all ACKs arrive before the expiration of the (short) SYNACK retransmission timeout. In such an interval also an early retransmitter would be located along the line $N_{SYNACK} \approx N_U$ in Fig. 5(a). The same scatterplot for connections via GPRS/EDGE is depicted in Fig. 5(b). The qualitative shape is comparable to Fig. 5(a). However, the existing clusters are less clearly separated. In Fig. 5(b) we observe points where $N_{SYNACK} \gg N_U$ also for servers not classified as early retransmitters. Note that this is not necessarily due to misclassification of the corresponding servers, since a SYNACK can be replied ambiguously also due to other effects than early retransmission of SYNACK.

3 Measurement Results

In the following we present three illustrative examples of abrupt changes in the network-wide performance signals (i.e. IS_{SYNACK} and RTT percentiles) found in an operational 3G network in Austria between June and October 2009. By investigating the root causes of these sudden deviations we will discuss relevant practical issues and show the applicability of these performance signals for detecting anomalies in a real 3G network.

3.1 Client-side RTT per Network Area

One interesting feature of the METAWIN monitoring system [11] is the analysis of TCP RTT, separately for different SGSN and RNC areas. In Fig. 6 we plot two timeseries of client-side RTT percentiles for connections established towards TCP port 80 and via UMTS/HSPA. The RTT percentiles for a specific

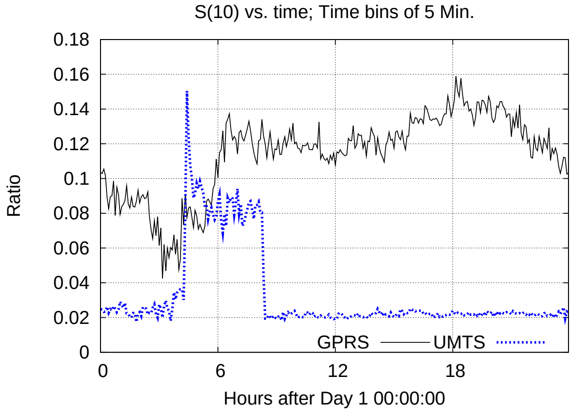


Fig. 7. Temporary increase of $S_L(10)$, 1 day time series, 5 min bins.

SGSN area are depicted in Fig. 6(a), while Fig. 6(b) shows the RTT percentiles measured via a RNC directly connected to the GGSN (see dashed line labelled "direct tunnel" in Fig. 1). In both cases the percentiles are relatively stable over time, showing statistical fluctuations during night hours when traffic load, and consequently also the number of RTT samples per measurement bin is low. As expected the direct path bypassing the SGSN has lower client-side RTT.

3.2 Temporary Increase of Packet Loss

In Fig. 7 we report a 24 hour timeseries of ISR separately for UMTS/HSPA and GPRS/EDGE. The estimated ISR of GPRS/EDGE shows a time-of-day variation between slightly aboc 0.04 in the night hours and around 0.16 in the peak hour after 18:00. However, we observe a sudden increase of the estimated loss probability $S_L(10)$ of UMTS/HSPA starting at 04:00 and lasting until around 08:00. This sudden shift from 0.02 to 0.08 with a distinct spike of around 0.15 is a clearly anomalous behavior. A deep exploration of the phenomenon at hand showed that this anomaly was caused by a temporary network problem associated to the reconfiguration of one GGSN, which led to partial packet loss at a specific site of the network. The presented anomaly is an important confirmation that, as we expected, an anomalous increase in packet loss is reflected by an abnormal deviation in the estimation of invalid sample ratio IS_{SYNACK}.

3.3 Detection of Bottleneck Link

Within our analysis of TCP RTT, we discriminate between server IP addresses allocated to the 3G operator (used for e.g. gateway servers, internal application servers) and all other IP addresses of the public Internet. The server-side percentiles for two consecutive days in time bins of 5 minutes only for internal

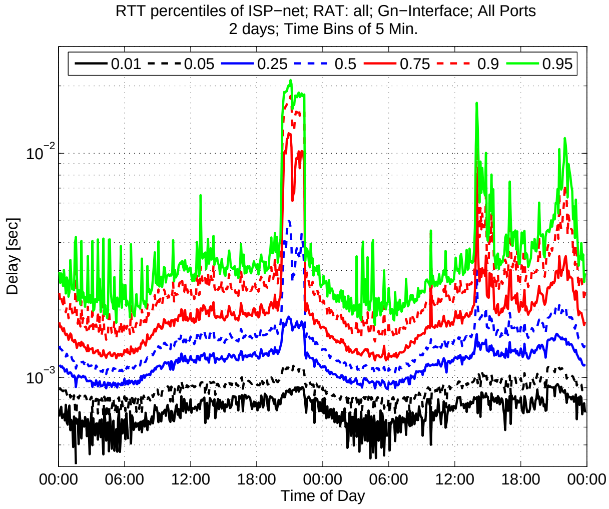
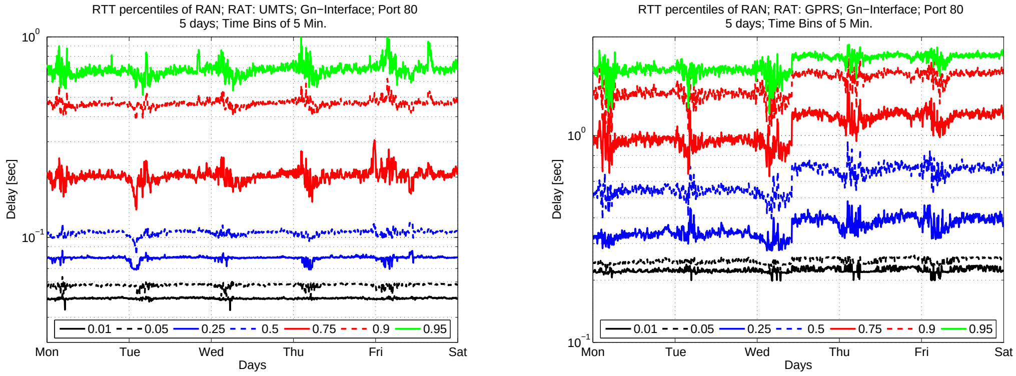


Fig. 8. Time series of server-side RTT percentiles, internal servers, 2 days, 5 min bins.

servers deployed by the mobile network operator are depicted in Fig. 8. The percentiles show a slight time of day effect, i.e. the server-side RTT is higher during the peak hours in the evening. This might be explained by two phenomena. First, an increase of traffic rate may lead to higher link utilization and thus larger delays on the path from/to the internal servers. Second, a higher load at the involved servers may increase their response time and hence also the server-side RTT. Besides a slight time-of-day effect of the percentiles we observe that 75% of the RTT samples take values below ≈2 ms. However, at around 20:30 of day 1 there is an abrupt change in the RTT percentiles. For instance, the 75-percentile is suddenly shifted from 2 ms to 10 ms. We observe that this shift of RTT percentiles is persistent for a period of about two hours. This is a clearly anomalous behavior. By taking into account also other signals obtained from the METAWIN system [11] we revealed that this RTT shift was contemporary to a significant increase of UDP/RTP traffic from the video streaming server during the live broadcast of a soccer match. In fact this traffic increase and consequently the abrupt shift in the server-side RTT was triggered by a significant number of users watching this live broadcast. Note the notch in the RTT percentiles at around 21:15 during the half-time break of the soccer match. Moreover, Fig. 8 shows a second abrupt change in the RTT percentiles at the second day with a clear spike around 14:00. Similarly to the example of the soccer match, this anomaly was caused by users watching the live broadcast of a Formula One race. Note that the increase of video streaming traffic did not only increase the server-side RTT towards the video streaming server, but also towards all internal servers located in the same subnet. Our findings finally pointed at a hidden congestion bottleneck on the path towards these internal servers of the network



(a) UMTS/HSPA

(b) GPRS/EDGE

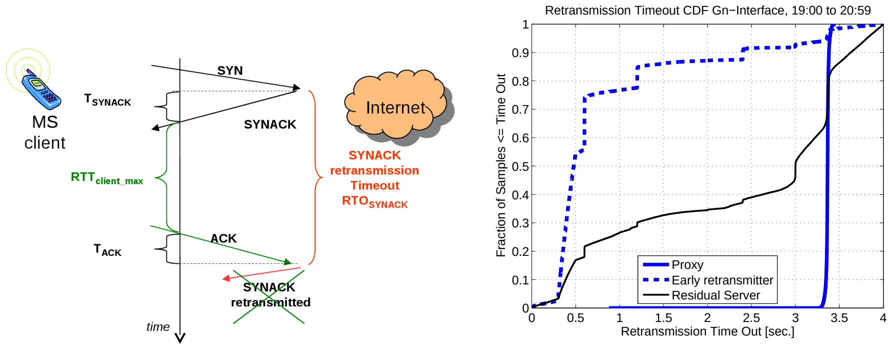
Fig. 9. Time series of client-side percentiles, 5 days, 5 min bins.

operator. After reporting our results to the network staff, the problem was fixed by increasing the capacity of the corresponding link. It is interesting to note that a traffic congestion caused at the UDP layer was discovered by analyzing just the handshaking packets of TCP at a single monitoring point. This result confirms the value of leveraging passively extracted TCP RTT for detecting bottleneck links in an operational 3G network.

3.4 Activation of Transparent Proxy

Fig. 9 depicts two 5 day time series of client-side RTT percentiles, separately for UMTS/HSPA and GPRS/EDGE. The client-percentiles of UMTS/HSPA are stable over time and do not show time-of-day variation. The median RTT is slightly above 100 ms. In the network under study, traffic load is steadily increasing during day hours (reaching its peak after 20:00) and decreasing again during night hours. The fact that client-side percentiles of UMTS/HSPA are independent from the actual traffic load suggests that the network under study is well-provisioned. Also the client-side percentiles of GPRS/EDGE in Fig. 9(b) are stable and independent from the variations of traffic load. However, Fig. 9(b) shows a sudden and persistent shift in client-side percentiles of GPRS/EDGE in the morning of the third day. For instance, the median of RTT is shifted from below 600 ms to around 700 ms. Further analysis revealed that the shift of RTT percentiles observed in Fig. 9(b) is caused by a reconfiguration of the network, specifically the activation of a network-wide proxy mediating TCP connections to Port 80 established in the GPRS/EDGE RAN.

For further clarification we now elaborate on the dependence of the client-side RTT on the SYNACK retransmission timeouts of remote servers. Recall from § 2.1 that we cannot compute a valid client-side RTT whenever a SYNACK is retransmitted by a remote server, since this retransmission leads to an ambiguous relation between two observed SYNACK and the ACK. A remote server



(a) Dependence of client-side RTT

(b) CDF RTO per server class

Fig. 10. Relation of SYNACK retransmission timeout and client-side RTT.

retransmits a SYNACK if it did not receive the client ACK within a specific retransmission timeout (RTO). In other words, we can compute a valid client-side RTT (inferred from an unambiguous SYNACK/ACK pair) if and only if a client ACK arrives at the remote server before the expiration of its SYNACK retransmission timeout referred to as RTO_{SYNACK} in Fig. 10(a). Let T_{SYNACK} denote the time required for a server SYNACK to arrive at our passive probe. T_{ACK} represents the delay a client ACK experiences between our monitoring point and the remote server. The maximum client-side Round-Trip-Time RTT_{client_max} that may be computed before a server is invalidating the current sample by retransmitting the SYNACK is then defined by

$$RTT_{client\_max} = RTO_{SYNACK} - (T_{SYNACK} + T_{ACK}). \tag{4}$$

We note that the time delay $T_{SYNACK} + T_{ACK}$ is equivalent to our definition of server-side RTT. In [3] we have shown that 75% of the server-side RTT values are smaller than 40 ms and 95% are below 200 ms in the network under study. This RTT is small compared to a SYNACK retransmission timeout of 3 seconds recommended in [14]. From Eq. 4 we observe that RTT_{client_max} is directly correlated to the setting of RTO_{SYNACK} at the server. In Figure 10(b) we plot a CDF of SYNACK retransmission timeouts inferred from SYNACK retransmissions within a time period of 2 hours from 19:00 to 21:00 for different classes of servers. The blue dashed line refers to the RTOs of servers being classified as early retransmitters. We note that 50% of SYNACKs were retransmitted after less than 500 ms and around 93% after less than 3 seconds. These short retransmission timeouts introduce a bias of the client-side RTT percentiles towards lower values, since RTT_{client_max} is decreased. In order to mitigate this problem we introduced a filtering of early retransmitters. In § 2.3 we have shown that the presence of early retransmitters introduce an error of 50% to our estimation of IS_{SYNACK}. Moreover, in Figure 10(b) we observe that residual servers retransmit 30% of SYNACKS after a timeout of less than 1.2 seconds and 45%

after less than 3 seconds (ref. black solid line in Fig. 10(b)). In contrast to that the network-wide proxy deploys only RTOs of above ≈3.2 seconds, which allows for a maximum measurable client-side RTT of around 3 seconds. This RTO setting of the network-wide proxy is the explanation of the persistent shift of the client-side RTT in Fig. 9(b). After its activation at day 3 the proxy increased the maximum measurable client-side RTT for all GPRS/EDGE users establishing TCP connections to port 80. This change was reflected in a larger relative share of higher client-side RTT values. In Fig. 9(a) we observe that there is no change in the client-side percentiles of UMTS/HSPA, since the proxy was only activated for users in the GPRS/EDGE RAN.

At this point it becomes obvious that client-side RTT percentiles are strongly dependent on the RTO settings of the involved remote servers. That means an abrupt change in the client-side RTT percentiles might be triggered not only by network problems, but also by a reconfiguration of the remote servers or the introduction of a transparent proxy as in the example of Fig. 9(b). One can even argue that not only early retransmitters, but all servers handling ports other than port 80 bias the RTT percentiles since also these servers deploy a large relative share of RTOs less than 3 seconds. For instance, one possible approach to mitigate this bias would be the filtering of handshakes of those servers, which retransmit at least one SYNACK after a timeout of less than 3 seconds. However, in the present example such filtering strategy would discard around 72% of all samples from our measurements while only 11.8% of samples are discarded by our filtering of a few early retransmitters. Hence, we believe filtering of early retransmitters is a trade-off between measurement accuracy and the completeness of the network-wide RTT measurements. In our work we compensate for the different RTO setting of the network-wide proxy by collecting and analyzing performance signals of two separate classes, one class referring to handshakes to port 80 (i.e. traffic mediated by the network-wide proxy) and a second class for all other ports.

4 Conclusions and Future Work

In this work we have investigated the possibility to exploit passively extracted TCP performance signals for anomaly detection in an operational 3G network. We have shown that some examples of real anomalies found in an operational network were reflected by abnormal deviations in the performance signals under study, i.e. invalid sample ratio and RTT percentiles.

Our results show that the RTT measured at the client-side does not only depend on the current network status, but also on the RTO setting of the remote servers and of intermediate proxies, if present. This calls for additional caution when relying on passive TCP RTT measurements for a comparison of different datasets — e.g. from different networks, or taken at different times. In fact, since RTO settings vary across different servers, variations in the global RTT distribution do not necessarily relate to network-level changes but might be caused by differences in the distribution of traffic mix and/or variations in the server popularity. We have proposed a dynamic classification and filtering of early retransmitters in order to mitigate their influence (bias) on the ISR and RTT measurements. We are aware that this classification and filtering involves additional computational efforts and reduces the complexity gap between the approach based on the analysis of handshaking packets exclusively and the approach of considering all DATA/ACK pairs. On the other hand, if RTT measurements are exploited for the purpose of anomaly detection, relying just on handshake packets yields a distinct advantage. Since the transmission delay of a packet depends directly on the packet length, variation of the packet length translates into statistical fluctuation of the packet delay — and hence of the associated RTT. By inferring RTT only from small handshake packets, we exclude these "normal" statistical variations which otherwise would complicate the task of detecting "abnormal" RTT deviations caused by anomalous events.

Our results show that the invalid sample ratio can be used as a complementary signal, e.g. to detect abnormally high levels of packet loss. Finally, thanks to our analysis of TCP RTT percentiles we have revealed one real instance of a link bottleneck caused by a temporary increase of UDP/RTP traffic. Taken collectively, our results show that such measurements can be fruitfully exploited in support of the operation and troubleshooting of a real-world 3G network.

Notably all the abrupt changes presented in the measured time-series could be easily detected by means of thresholding and/or very basic change-detection algorithms. In this study, the main challenge is not on the design of sophisticated signal processing algorithms for time-series, but rather on the extraction of a robust and reliable network signal to base the detection upon. As part of ongoing work, we are now integrating basic alarming thresholds for such signals into the on-line monitoring system.

Acknowledgments

The Telecommunications Research Center Vienna (FTW) is supported by the Austrian Government and the City of Vienna within the competence center program COMET. This work is part of the DARWIN project [11] run at FTW.

References

1. P. Benko, G. Malicsko and A. Veres: A Large-scale, Passive Analysis of End-to-End TCP Performance over GPRS. IEEE INFOCOM 2004.
2. F. Vacirca, F. Ricciato, R. Pilz: Large-Scale RTT Measurements from an Operational UMTS/GPRS Network. Proc. of WICON'05, Budapest, July 2005
3. P. Romirer-Maierhofer, F. Ricciato, A. D'Alconzo, R. Franzan, W. Karner: Network-wide measurements of TCP RTT in 3G. 1st International Workshop on Traffic Monitoring and Analysis (TMA09), Aachen, Germany, May 2009.
4. J. Aikat, J. Kaur, F.D. Smith, K. Jeffay: Variability in TCP round-trip times. ACM SIGCOMM IMC 2003, Miami Beach, USA, October 2003.

5. S. Jaiswal, G. Iannaccone, C. Diot, J. Kurose, D. Towsley: Inferring TCP Connection Characteristics Through Passive Measurements. IEEE INFOCOM 2003, San Francisco, USA, April 2003.
6. S. Rewaskar, J. Kaur, F.D. Smith: A passive state-machine approach for accurate analysis of TCP out-of-sequence segments. ACM SIGCOMM Computer Communication Review, vol. 36, n. 3, pp. 51–64, July 2006.
7. M. Mellia, M. Meo, L. Muscariello, D. Rossi: Passive analysis of TCP anomalies. Computer Networks, vol. 52, n. 14, pp. 2663-2676, June 2008.
8. RFC2988: Computing TCP's Retransmission Timer. November 2000.
9. F. Ricciato, F. Vacirca, P. Svoboda: Diagnosis of Capacity Bottlenecks via Passive Monitoring in 3G Networks: an Empirical Analysis. Computer Networks, vol. 51, n. 4, pp. 1205-1231, March 2007.
10. J. Bannister, P. Mather, S. Coope: Convergence Technologies for 3G Networks: IP, UMTS, EGPRS and ATM. Wiley, 2004.
11. METAWIN and DARWIN projects: `http://userver.ftw.at/~ricciato/darwin`
12. "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface," *3GPP TS 29.060, Version 8.9.0, Release 8*, October 2009.
13. A. Coluccia, F. Ricciato, P. Romirer-Maierhofer: On Robust Estimation of Network-wide Packet Loss in 3G Cellular Networks. IEEE BWA'09, Honolulu, USA, 30 November 2009.
14. RFC1122: Requirements for Internet Hosts - Communication Layers. October 1989.