

# Bottleneck Detection In UMTS Via TCP Passive Monitoring: A Real Case

Fabio Ricciato and Francesco Vacirca  
Telecommunications Research Center Vienna  
Donau City Strasse 1,  
A-1220 Vienna, EU  
{ricciato,vacirca}@ftw.at

Martin Karner  
mobilkom austria AG & Co KG  
Obere Donaustrasse 29,  
A-1020 Vienna, EU  
martin.karner@mobilkom.at

## ABSTRACT

In this work we address the problem of inferring the presence of a bottleneck from passive measurement in the UMTS core network. The study is based on one month of packet traces collected in the core network of mobilkom austria AG & Co KG, the leading mobile telecommunications provider in Austria, EU. During the measurement period a bottleneck link in the UMTS core network was revealed and removed, therefore the traces enable the accurate analysis and comparison of the traffic behavior in the two network conditions. The proposed approach exploits statistics of estimated TCP performance parameters (e.g. RTT, re-transmissions) in order to build a set of bottleneck indicators. We show that such statistics are volatile due to the presence of few top users, but this effect can be counteracted with a simple filtering method. Results show that the frequency of re-transmission events is a powerful indicator for the specific type of bottleneck under study, and it can be used to provide early warning about future occurrences of similar events. This application is particularly important for operational UMTS networks nowadays, since the traffic volumes and composition is still under evolution.

## Categories and Subject Descriptors

C.2.1 [Computer Communication Networks]: Network Architecture and Design

## General Terms

Measurement, Performance

## Keywords

Bottleneck Detection, UMTS

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CoNEXT'05, October 24–27, 2005, Toulouse, France.  
Copyright 2005 ACM 1-59593-197-X/05/0010 ...\$5.00.

## 1. MOTIVATIONS

Public wide-area wireless networks are now migrating from second-generation systems (2G) developed for low-bandwidth circuit-switched services, towards third-generation systems (3G), designed to support higher data rates and packet-switched services. Several 3G systems are being developed evolving from different 2G technologies. European countries have adopted the Universal Mobile Telecommunication System (UMTS), developed by 3GPP as an evolution of GSM. During the migration path from GSM to UMTS, an intermediate phase is General Packet Radio Service (GPRS), one of the so-called 2.5G technologies. Several 3G operators maintain two parallel Radio Access Networks (RAN), GPRS and UMTS, sharing a single Core Network (CN). The CN is packet-switched and largely based on IP-based protocols. A general overview of the GPRS/UMTS network structure can be found in [7].

Several UMTS networks became operational since 2003<sup>1</sup>, while the first deployments of GPRS date back to 2000. The development of the GPRS/UMTS network structure runs in parallel with the evolution of terminals (or Mobile Stations - MSs). Evolved cellphones and smart-phones already support a broad range of data applications, including traditional Internet applications like e-mail, WEB, etc. Besides handheld terminals, 3G interface cards for laptop started to be commercially available in Europe in fall 2004, often coupled with flat-rate connectivity contracts, and met a considerable market success. Since then, the volume of traffic generated by 3G terminals and directed to the Internet has registered a steady increase.

The growing popularity of 3G data services has extended the coverage of Internet access to the geographic area, and 3G networks are becoming key components of the global Internet in Europe. However, 3G networks and markets are still under evolution, and changes occur rapidly: the subscriber population and the traffic volumes are still in a growing phase; the relative distribution of terminal types (e.g. laptops vs. handsets) and their capabilities is changing quickly; the portfolio of services that are offered by the operators and/or spread popular among the customers evolves rapidly and is still far from stabilization. In addition to that, prospective changes to the network structure are in the agenda of many operators, including capacity increases, new software releases, technology upgrades (e.g. EDGE and IMS [7, pp. 155 and 555-560]).

<sup>1</sup>A detailed list can be found at <http://www.gsmworld.com/technology/3g/index.shtml>

All these aspects concurrently build a potential for changes in the traffic patterns that can occur at the macroscopic scale (network-wide) and in a relatively short time frame. In such a scenario, the ability to accurately and extensively monitor the current network state and to early detect drifts in the network performance is not just a useful complement, but a fundamental pillar of the network operations and engineering processes. On the other hand, monitoring a wide-area network involves considerable costs, particularly in the radio access section. The number of links to be monitored is large, and they are spread over a wide geographical area. For some operators there are several parallel RANs to be monitored that are attached to the same CN: UMTS, GPRS and WLAN hot-spots (see[8] for 3G-WLAN interworking), with the possibility of yet additional technologies to be introduced in the future (e.g. WIMAX). For some monitoring applications it is required to access configuration parameters (e.g. provisioned link bandwidth), logs and built-in counters from several network elements, and any experienced engineer in the field is well aware of the cost, complexity and complications that are found in practice where it comes to extraction, gathering and correlation of such heterogeneous data from different elements, with different SW releases and from different vendors. In summary, installing and maintaining a monitoring infrastructure with the same capillarity of the production network might be tremendously expensive. Fortunately, the structure of a 3G network is highly hierarchical and centralized: the whole traffic is concentrated in the Core Network and there are only few gateway nodes - called GGSNs - that connect the 3G network to external networks like the Internet. It would be highly desirable for the network operator to be able to infer the presence of performance bottlenecks anywhere in the network - included the Radio Access Network - monitoring the traffic only at few capture points near the GGSNs. A possible approach to achieve this goal is to look at TCP behavior: since TCP is closed-loop controlled, its dynamics and performances are dependent on the state of the whole end-to-end flow path. In principle it should therefore be possible to infer the presence of bottlenecks by looking at the evolution of the TCP aggregate and/or to individual connections at any point along the path.

A point of clarification is due regarding the definition of “bottleneck” in this context. In a well-engineered 3G network the traffic flow of each MS is naturally rate-constrained by the availability of bandwidth on the radio channel. The radio bandwidth is shared between the MSs active in the same cell, therefore the bandwidth available at any instant to individual MSs varies with the total offered traffic, in addition to the variability due to physical factors (e.g. fading, SNR variability). Ideally the radio planning is such that on average each MS accesses a sufficient level of average bandwidth during its activity period, and the capacity of the core network links can always sustain the traffic aggregate without packet loss nor any other performance degradation. This is the ideal *modus operandi* of a network: “bottleneck free”. The network state might depart from such condition due to changes in the volume or distribution of the traffic that were not anticipated by the network designer. If the traffic reaches the capacity limit of some network element  $X$  causing performance degradation to the traffic flows, and this happen for long periods and recurrently over multiple days, we say that a capacity bottleneck has emerged on  $X$ .

Its removal implies capacity upgrades at the specific network element. Note that a sporadic shortage of capacity e.g. due to a flash crowd is not accounted as a bottleneck since it does not necessarily imply long-term capacity re-assignment. A bottleneck in the radio network can be a geographical region that is frequently overloaded for long periods because of an inadequate assignment of radio capacity in that area. A bottleneck in the core network is often a link with too little capacity to carry peak-hour traffic.

From the above definition it derives that a capacity bottleneck always impacts a certain traffic aggregate rather than isolated flows - e.g. all the traffic directed to a certain radio area, or routed over a certain network element - and inevitably degrades the performance of several flows at the same time.

We devised two possible approaches for inferring the presence of a bottleneck on network element  $X$  from the analysis of the traffic aggregate routed through it:

- From the statistical properties of the aggregate traffic rate: if the traffic rate is closed-loop controlled - as is the case of traffic mix with large prevalence of TCP - the aggregate rate will adapt itself to the capacity limit of the bottleneck, if this is in place. In this case it can be expected that the statistical properties of the aggregate rate will be different from the normal bottleneck-free conditions, therefore they can be used to discriminate between the two states.
- From the performances of the TCP connections within the aggregate (re-transmission events, RTT, etc.).

In this paper we consider only the latter approach, with reference to a real case found in practice. This study is based on one month of packet traces collected by passive monitoring the links of the 3G core network of mobilkom austria AG & Co KG, the leading mobile telecommunications provider in Austria, EU. In the middle of the measurement period a bottleneck link within the UMTS Core Network was detected and removed. Therefore, our traces enable the accurate analysis and comparison of the traffic behavior before and after the removal. Since the traces are complete - i.e. all packet headers have been captured and timestamped - it is possible to analyze any aspect of the traffic dynamics that might be of interest. The goal is to identify those patterns that more clearly discriminate between the two network conditions: with and without the bottleneck in place. This approach is similar to the so called “post-mortem” analysis that is commonly performed in the area of network security in order to learn about the dynamics of past attacks and devise countermeasures and detection mechanisms. The latter are typically based on specific patterns associated to the attack, often called “signatures”. While in network security a signature is often a specific content in the packet payload (e.g. a piece of malicious code), in our case the signature of a bottleneck is to be found in some statistical pattern extracted from the traffic process (e.g. the frequency of retransmissions, or some moment of the marginal distribution of the aggregate rate). However the underlying principle is the same: monitoring and storing the traffic that occurred during an undesired event (an attack or, in our case, the presence of a bottleneck), analyzing in post-processing what had happened (“post-mortem analysis”) and comparing with traffic patterns in “normal”

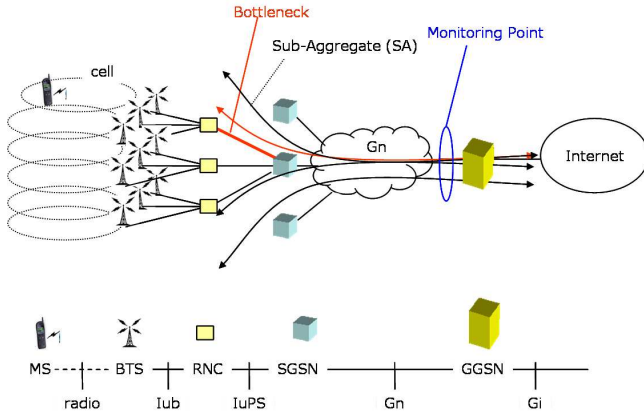


Figure 1: Reference network scenario

operating conditions, i.e. identifying one or more “signatures” of the events. The practical application of this study will be to implement an intelligent testing agent to be integrated into the on-line monitoring system, watching for future occurrences of similar patterns and reporting early warning about future bottlenecks.

The rest of the paper is as follows. In Section 2 we present the reference network scenario and state the problem. Section 3 describes the monitoring setting and the traces. In Section 4 we present the measurements results. Finally in Section 5 we review the literature related to our work and in Section 6 we draw the conclusions and identify directions for future work.

## 2. REFERENCE NETWORK SCENARIO AND PROBLEM STATEMENT

The reference network scenario is depicted in Figure 1. The 3G network has a tree-like deployment: the terminals and base-stations are geographically distributed (e.g., nation-wide), but the level of concentration increases when moving towards the boundary of the 3G network towards the Internet. There are in general a few number of SGSNs and even fewer GGSNs operational in each network, and the traffic is concentrated on a small number of Gn/Gi links, therefore with a few probes one is able to capture the entire traffic aggregate on these interfaces.

The problem addressed here is how to infer the presence of a bottleneck in the 3G network, between the Gn and the radio cells, from the passive observation of the traffic at a single monitoring point - on Gn in our case. Remarkably, we assume only a minimal information about the structure and settings of the whole network: for instance, we assume that the **bandwidth provisioned at each link is not known** to the monitoring agent, nor the detailed network topology is known. The only required information are those enabling the discrimination of different sub-aggregate components inside the total aggregate. A sub-aggregate (SA for short) defines a portion of the overall traffic that is routed through a specific part of the network. A SA can be associated to each network element (e.g. SGSN, RNC, Routing Area, Cell). For example, one could define a SA for a single SGSN  $x$ , meaning that all the traffic routed through SGSN  $x$  can be separated by the rest and examined separately.

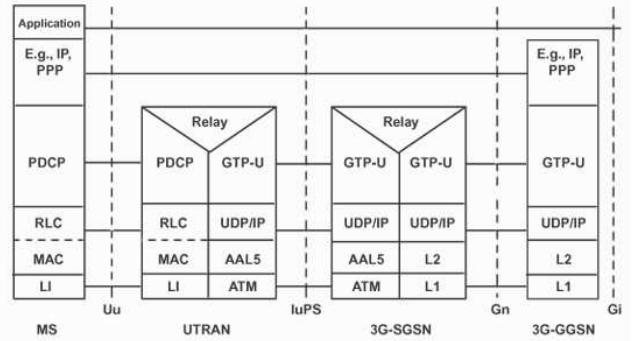


Figure 2: UMTS protocol stack (data-plane).

At a coarser granularity one could define a SA for a SGSN cluster  $y$ , typically including all SGSNs co-located at a single physical site. At a finer granularity one might consider all the traffic to a single RNC (say  $z$ ). An even higher granularity, e.g. per-cell, might be used for detecting recurrent bottlenecks or heavy congestion in the radio link, and then trigger a local revision of the radio planning. Due to the typical tree-like structure of the 3G network SAs are hierarchically nested (in our case  $z \subset y \subset x$ ), and the analysis at different depth in the hierarchy might help to individuate the position of a bottleneck, an approach that is in principle similar to network tomography [3].

Sub-aggregate analysis requires discrimination, that is the ability to refer each traffic unit (a packet or a connection) to a specific SA. In practice, the way how this association is implemented depends on several technical details that are often dependent on the specific configuration of the network. Under this respect, it is preferable to set the monitoring point on Gn, since the lower layers of the Gn protocol stack (see Figure 2) include several useful information for this purpose. The Gn network is IP-based: after GTP encapsulation (see [7, pp. 84-85] for details) the user packets are encapsulated into UDP/IP packets carrying the IP address of the destination SGSN/GGSN; this allows direct per-SGSN and per-GGSN discrimination. A higher level of discrimination can be achieved by tracking the cell/routing-area information present in some messages exchanged between the MS and the GSNs. The detailed tracking mechanism is different for GPRS and UMTS. In this paper we do not consider further the technicalities involved in the task of SA discrimination, since this would require a thorough treatment of the 3G protocols and some insight into the engineering practice of a real network. As a working hypotheses we assume that the monitoring system is capable of capturing and discriminating all packets belonging to a certain SA, and the problem is then to infer the presence of a bottleneck affecting such specific SA.

In the most simple approach, the analysis of each SA is performed in isolation. In other cases the comparison between the dynamics of different SAs at the same hierarchical level might be useful to pinpoint an abnormal behavior. However a point of caution is due here: different SAs might behave differently because of different load conditions, independently from the presence of any abnormal bottleneck. Consider for example that the traffic to different SGSNs is

generated by users in different geographical areas, with e.g. SGSN  $x_1$  covering an urban area and SGSN  $x_2$  the neighbor rural area, holding very different traffic mix and volume from each other. Therefore a difference in traffic dynamics of  $x_1$  and  $x_2$  does not necessarily point to an anomaly or bottleneck. Still, if one is able to extract summary indicators that are invariant with different load conditions, it would be possible to detect different operational conditions from direct comparison of parallel SAs. The identification of synthetic and invariant indicators is one of the ultimate goals of our research.

### 3. MONITORING SETTING

The development of a large-scale passive monitoring system - including a parser for the whole protocol stack of the 3G Core Network - and its deployment in the operational network were accomplished within the METAWIN project [11]. Packets are captured with Endace DAG cards [1] and recorded with GPS synchronized timestamps. For privacy requirements traces are anonymized by hashing any field related to user identity at the lower layers of the 3G stack (IMSI, MSISDN, etc.). They include TCP/IP headers enabling the analysis of several TCP statistics. While we passively monitor all core network interfaces (Gi, Gn, Gb, IuPS) the results presented in this work are based exclusively on Gn traces. All Gn links were monitored, covering 100% of GPRS and UMTS traffic from home subscribers, traffic of roaming subscribers is not considered.

For this work we collected more than 4 weeks of traces during November-December 2004. A bottleneck was in place in the network and was removed during the measurement period. The bottleneck affected only a certain portion of the UMTS traffic: we were able to discriminate the SA component crossing the bottleneck out of the total Gn trace and analyze it separately. In the rest of the paper we will therefore refer exclusively to the analysis of this sub-trace.

For proprietary reasons we can not disclose several absolute quantitative values (e.g., traffic volumes, number of users, number of Gn links, etc.) nor any other information that might indirectly lead to them (e.g., absolute number of RTT samples). For these quantities we will provide only relative values, i.e. fractions, rather than absolute ones.

As discussed above, we are interested in looking at TCP behavior because of the end-to-end nature of its dynamics. Our measurements confirmed a large prevalence of TCP traffic in the 3G network. Furthermore it is likely that a large part of the traffic seen as UDP packets in the Core Network is accountable as TCP-controlled: for instance TCP connections tunneled into IPsec VPNs will be seen as UDP at our monitoring point. The measured traffic is highly asymmetric (the largest part of the volume was WEB), with an average uplink:downlink ratio of approximately 1:3. Since the provisioned bandwidth for the CN links was symmetric, only the downlink traffic reached the bottleneck capacity.

In fig. 3 we plotted the volume of the SA under study during one week in the measurement period, corresponding to day 17-23. Only the downlink traffic is plotted. Individual samples refer to time bins of 10 sec, and the embedded curve shows the 1 hour moving average. The values have been normalized to an arbitrary value  $u$  in order to not disclose the absolute volume of traffic. A bottleneck with bandwidth limit  $\beta = 0.5u$  was in place until the middle of the week, and was removed in the night between day 20 and 21.

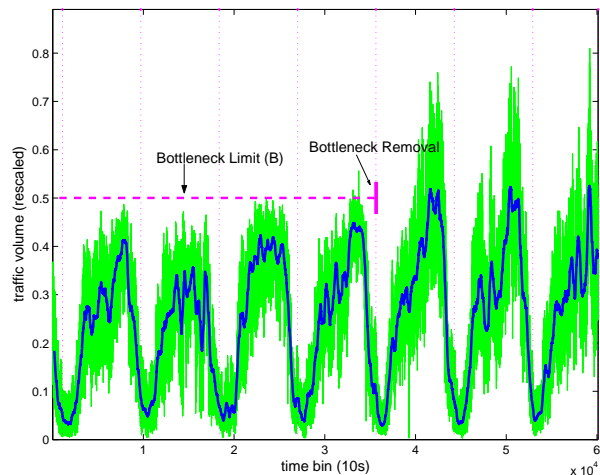


Figure 3: Total SA traffic volume from day 17 to 23, downlink byte count, 10s bins (rescaled to arbitrary unit).

## 4. ANALYSIS OF TCP PERFORMANCE INDICATORS

### 4.1 Method

We considered several TCP performance parameters as candidate indicators for the presence of a bottleneck. The first group of potential indicators refers to the frequency of re-transmission events, discriminated into the following categories:

- **FRTX** (Fast-Retransmit Re-transmissions): the re-transmission of a packet triggered by duplicate ACKs.
- **LRTO** (Loss-induced Retransmission Time-Outs): a packet re-transmission triggered by the expiration of the TCP Retransmission Time-Out (RTO) caused by packet loss.
- **SRTO** (Spurious Retransmission Time-Outs): the occurrence of a re-transmission due to RTO expiration caused by a large delay, without packet loss.
- **AMB** (Ambiguous Retransmission Time-Outs): the occurrence of a re-transmission that could not be classified into one of the previous categories. Note that AMB events are always associated to RTO expiration.

All these events were inferred with the procedure proposed in our previous work [6]. Therein the focus was on the estimation of the SRTO events, which in turn required the discrimination of LRTO, FRTX and AMB. We have implemented the estimation algorithms in a modified version of the `tcptrace` tool [2], which was run over the whole traces<sup>2</sup>. For each type of event we measured the relative frequency into each time bin as follows. For each active terminal  $i$  we counted the number of occurrences of the specific event (e.g. SRTO)  $n_{S,i}$ , and the total number of DATA packets

<sup>2</sup>The modified `tcptrace` version can be downloaded from <http://userver.ftw.at/~vacirca/>

$N_i$ . The global frequency  $f_S$  is then defined as :

$$f_S = \frac{\sum_i n_{S,i}}{\sum_i N_i} \quad (1)$$

In this work we used the IP address on the MS side as an identifier for the MS. In the 3G network IP addresses are dynamically assigned to MSs when they connect to the network (PDP-context activation [7, p.413]) and released when the connection (PDP-context) is deactivated. After that in principle the same IP address might be immediately re-assigned to another connecting MS. However we had the means to check off-line that in the network under monitoring the level of address re-usage by different MS within 1 hour is almost negligible, i.e. only a very small fraction of IP addresses were used by two or more different terminals within each time bin. On such basis we assign each packet to a MS, accepting the small error caused by short-term address re-usage.

In addition to the above events we also considered the Round Trip Time (RTT) values. A first exploratory analysis of TCP RTT in UMTS and a comparison with GPRS was reported in our previous work [5]. Instead of the end-to-end RTT, we considered the semi-RTT between the monitored interface (Gn) and the Mobile Station (MS). For sake of simplicity, in the rest of this work we will refer the Gn-MS-Gn semi-RTT simply as “RTT”. An RTT sample is defined as the elapsed time  $t_{data} - t_{ack}$ , where  $t_{data}$  and  $t_{ack}$  are the timestamps respectively of a TCP DATA packet arriving from the Internet and of the associated ACK from the MS as “seen” at the monitoring point on Gn. The RTT defined in this way includes three components: the down-link delay (Gn→MS), the uplink delay (MS→Gn) and delay component internal to the MS (e.g. processing and I/O buffering). Only the first two components are accountable as network-dependent, while the latter depends exclusively on the terminal. However, the TCP dynamics, and ultimately the user-perceived performances, will be impacted by the cumulated RTT. Note that only non-ambiguous DATA-ACK pairs are considered to produce a valid RTT sample: the acknowledgment number of ACK must be at least one byte greater than the last sequence number of the DATA packet; furthermore, it is required that the packet being acknowledged was not retransmitted, and that no packets that came before it in the sequence space were retransmitted after  $t_{data}$ . The former condition invalidates RTT samples due to the retransmission ambiguity problem. This is the same procedure that TCP utilizes to estimate the RTT and to set the value of the Retransmission Timeout (Karn’s algorithm), see [19]). The latter condition invalidates RTT samples since the ACK could acknowledge cumulatively the retransmitted packet rather than the original DATA packet.

Before the analysis we filtered out all packets on ports tcp:4662 and tcp:445/135. The former is used by popular peer-to-peer file sharing applications: since it typically runs with many parallel TCP connections it is likely to induce self-congestion on the radio channel and/or on the terminal internal resources (e.g., transmission buffer). This would result into poor TCP performances that are application-specific rather than network dependent, therefore do not carry information about the network state. Additionally, during the exploratory analysis we found the presence of a large number of packets directed to ports tcp:445/135, mainly TCP SYN in the uplink direction. This is likely due

to some self-propagating worms attached to infected 3G terminals. The presence of such unwanted traffic should be expected since laptops with 3G datacards - often equipped with popular operating system - populate the 3G networks nowadays along with handsets and smartphones. It is well-known that unwanted traffic is a steady component of the traffic in the wired networks since years (see for instance [15]). The detailed analysis of such traffic and its impact on the 3G network will be covered in a following separate paper. What is important here is that most of such packets did not bring any valid contribution for the problem of bottleneck detection while consuming resources in the analysis software (i.e., memory state in `tcptrace`), therefore filtering them out speeds up the analysis process.

## 4.2 Results

In Figure 4 we plotted the measured values for each parameter in time bins of 1 hour. The top subgraph shows the cumulative number of TCP DATA packets ( $\sum_i N_i$ ), normalized to the peak value during the entire monitoring period in order to not disclose the absolute value. The second subgraph reports the average and several percentiles (5%, 50% 95%) of the RTT samples extracted in each time bin. The remaining subgraphs report the measured frequency of FRTX, LRTO, SRTO and AMB respectively. Recall that the bottleneck was removed in the night between day 20 and 21. From Figure 4 it appears that the RTT statistics display a large variability, with average values occasionally very large. However, a deeper look would reveal that most of the RTT spikes occur over night, when the traffic volume and the number of active terminals are very low. This suggested that such spikes might be the effect of few terminals generating a large volume of packets and hence RTT samples. In case that such packets are associated to large RTT values for some specific reason (e.g. poor local radio condition, intensive mobility, application-specific ACK delay), they introduce a bias in the whole RTT statistics. This is more likely when the network load - therefore the number of terminals - is low.

Regarding the other parameters, it is evident that the FRTX, SRTO and AMB frequencies display periodic spikes before day 21, particularly at the peak hour, which are clearly an effect of the bottleneck. Among them, the SRTO seems to be the best indicator for this type of bottleneck. In fact, it can be seen that after the bottleneck removal the SRTO frequency stays at a “physiological” level (below 0.1% in UMTS) that is highly stable: no large fluctuations are present, and there is no apparent dependency on the time-of-day and therefore on the network load. In other words, the “normal” value of SRTO frequency in UMTS is invariant to changes in the network load. As discussed above in Section 2, this is a highly desirable characteristic for a parameter that should be used as an indicator for abnormal network conditions.

The behavior of FRTX and AMB shows clearly some correlation with the presence of the bottleneck, with large spikes mostly during peak-load hours, but they also display some sporadic high values after the bottleneck removal. For the LRTO there are no evident differences in the behavior before and after day 21. However, similarly to the RTT, most of the spikes seen for FRTX, LRTO and AMB after the bottleneck removal are placed at off-peak hours, which again suggests the possibility of bias from a few top-outliers terminals.

In order to counteract the biasing effect we used the following simple approach to filter out outlier terminals. For the RTT, in each time bin we rank the terminals w.r.t. to the product of the average RTT measured  $\bar{r}_i$  and the number of valid RTT samples  $K_i$  (different from the number of DATA packets  $N_i$ , since only selected DATA-ACK pairs hold valid RTT samples). We filtered out the top 10 MSs with the highest  $\bar{r}_i \times K_i$  product in each time bin, and re-compute the RTT parameters (average and percentiles) from the residual set. For the other indicators we follow a similar approach: for each class of re-transmission event and for each time bin we filter out the top 10 MSs with the highest number of occurrences (e.g.  $n_{S,i}$  for SRT0), then recompute the total frequency  $f_S$  on the residual set of MSs. Despite of the use of a simple heuristic, our results show it is extremely well suited for our purposes. Further refinements are certainly possible, for instance the fixed number of filtered MS is definitely a limit of this approach, and we consider them a point for further study.

In Figure 5 we plot the same quantities of Figure 4 after the filtering process (note that the respective subgraphs might be in different scales). All the frequencies of re-transmission events now display a more predictable behavior after the bottleneck removal, thus confirming our hypothesis that most of the volatility was due to a few top-outlier terminals. The large residual spikes, now regularly located in the peak hours, disappear completely after day 21. This clearly relates with the presence of the bottleneck, and the change in the behavior is so clear that one can immediately pinpoint the bottleneck removal time.

Regarding the RTT statistics, the filtering process had a dramatic effect and almost completely canceled the fluctuations of the average - now firmly anchored to the level of 500ms - and of the lower percentiles. However, there is no evidence of correlation between the RTT values shown in Figure 5 and the presence of the bottleneck. The conclusion is that the RTT process, at least as estimated with the methodology described above, is not a good indicator for this type of bottleneck. The likely explication is that the RTT estimation process implemented in `tcptrace` only considers selected DATA-ACK pairs that do not hold any ambiguity in the RTT estimation. This method filters away “invalid DATA-ACK pairs”, that typically emerge in the neighborhood of events like packet loss, retransmissions and timeouts. Now, these are exactly the events that are generated by the bottleneck. In other words, RTT statistics were intrinsically “cleaned-up” due to the way RTT samples are extracted, which explains why they did not react to this type of bottleneck. Still, it can be argued that different types of bottleneck - e.g. those with large buffering space, as a shaper - might be better captured by indicators associated to the packet delay rather than to retransmissions or time-outs.

### 4.3 Considerations

The results carried by the above analysis show that it is possible to build powerful bottleneck indicators from performance parameters estimated by passive monitoring TCP traffic. The “goodness” of an indicator depends on its **predictability** under nominal operational conditions, which in turn requires limited volatility, and on its **responsiveness** to abnormal conditions. Once that a “good” indicator of the network state is defined, statistical testing methods can

be applied to provide automatic alarms when an abnormal condition occurs. From a general point of view, the “better” is the indicator signal in terms of predictability and responsiveness, the simpler can be the statistical testing technique.

In the specific case considered in this work we found that all the proposed indicators associated to re-transmission events are so effective that even the most simple testing method, i.e. a fixed threshold placed adequately set, would have provided early warning about the presence of the bottleneck several days in advance.

## 5. RELATED WORKS

Several papers have analyzed the characteristics of TCP connections in real networks. Most of them focused on traffic modeling purposes (e.g. [14]), whereas only few works exploited passive TCP measurements to infer the status of the network. In [17] the authors use heuristics based on the TCP congestion control mechanisms and on the estimation of the Re-transmission Time-Out in order to reconstruct the TCP sender behaviour, and classify out-of-order packets according to different causes. The same authors use in [16] a passive measurement methodology to infer the TCP sender congestion window and the connection RTT. As pointed out in both papers, their analysis is meant to be a first step towards a better understanding of the correlation between the properties of the traversed path and TCP connection specific metrics, ultimately aiming at the identification of the Autonomous System that degrades the connection throughput with high packet losses. While the underlying idea is similar to our work, namely exploiting TCP dynamics to infer performance degradation points, the application scenarios are different (Tier-1 backbone vs. 3G Core Network) and require different approaches and methodologies.

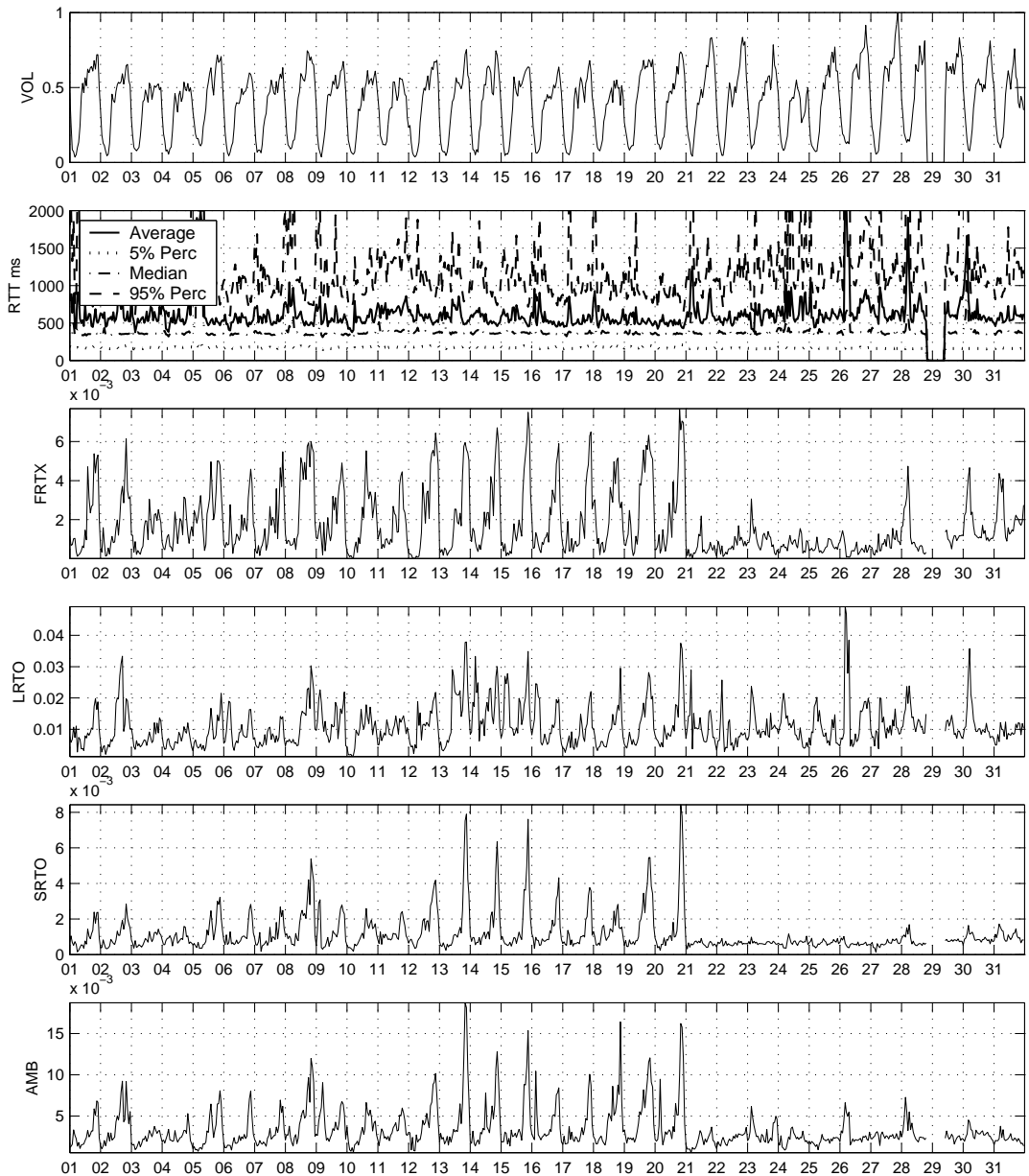
A validation of TCP performance over GPRS by investigating traces collected by passive monitoring at the Gi interface is shown in [12]. GPRS measurement results are compared to a wireline dial-up network problem.

In [10], the authors present the `Tstat` [9] tool for inferring the status of TCP connections and several associated metrics. Based on that they analyze the traffic on the edge of a campus network and estimate the performance from the perspective of individual users. Besides the different application scenario, the goal of their study is different from the present work. Our goal is to validate the current network state from the perspective of the operators and detect early warning about large-scale performance bottleneck.

Other papers have worked on the concept of detecting network anomalies from passive measurements, covering a broad range of proposed approaches (e.g. signal processing techniques [13]) or directly inferring capacity limits along a path (e.g. [18] used the analysis of packet inter-arrival patterns). To the best of our knowledge this is the first work that directly addresses the problem of bottleneck detection in 3G networks, and the first to report large-scale measurements from an operational network over several weeks of traces. Also, no previous paper so far has based this type of analysis on the empirical observation of a real bottleneck found in an operational network.

## 6. CONCLUSIONS AND ONGOING WORK

In this work we have suggested a possible approach to bottleneck detection based on passive traffic monitoring. The



**Figure 4: TCP performance indicators (1h bins). From top to bottom: total packet count, RTT, FRTX, LRTO, SRTO, AMB.**

method proposed here based on TCP performance indicators complements other methods that are currently under study and were left out of the scope of this paper.

A point of caution is due regarding the generalization of the results: it can not be expected that the detection mechanism developed out of a specific event will necessarily capture *all* possible types of bottleneck. In fact, TCP traffic might react differently to different forms of resource limitation: consider for example that a rate-limiter can be instructed to drop the packets in excess to the configured rate, or rather buffer them (shaper). In the two scenarios the presence of the bottleneck will likely appear on different param-

eters: loss and retransmission events in the former, larger delays and RTT in the latter. In other words, the physical nature of the bottleneck plays a role in defining its impact on the traffic and ultimately its “signature”. Therefore, we believe that bottleneck detection schemes to be used in production networks should be based on a bank of parallel tests on different indicator signals and parameters, in order to increase the likelihood that future events are not missed. Also, as new types of events are found during the network operation and their signature are investigated (“post-mortem analysis”), the set of detection algorithms can be extended with new tests, in a sort of continuous learning process that

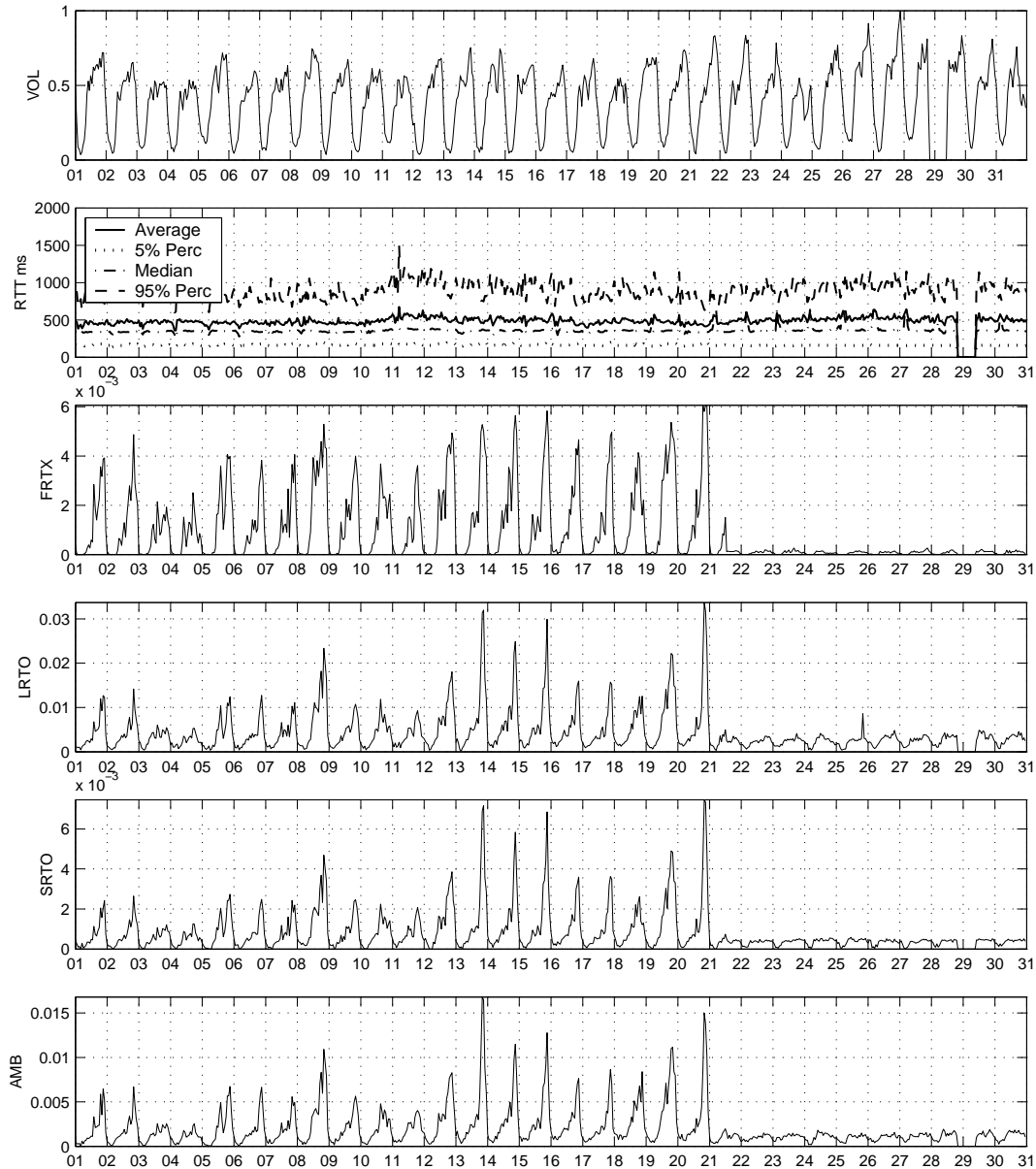


Figure 5: TCP performance indicators filtered from top-10 outliers flows.

is principle similar to the way Intrusion Detection Systems (IDS) and Antivirus software are evolved. The prerequisite for this type of study is the availability of complete packet traces.

We are currently implementing these methods on top of the on-line monitoring system developed in our project [11], and they will be activated in the production network of mobilkom austria.

The natural continuation of this work is to devise a framework for locating the performance bottlenecks from the comparative analysis of the SAs at different hierarchical levels (sites, SGSN, RNC, Cell). This approach holds some similarities to network tomography [3]: the common idea is to infer the internal state of the network from external measure-

ments (“black-box” approach). However we consider a different strategy since we base on passive measurements taken at a single capture points rather than active delay measurements between multiple sites. This is enabled by the tree-like structure of the 3G network, and relies on the fact that most of the traffic is TCP-controlled. However, the analysis at lower levels of granularity (e.g. per-cell SA) involves additional challenges, for instance the practical complications of discriminating fine-grain SAs. The hinger volatility associated to the lower level of traffic aggregation will probably require different testing techniques. Nevertheless, this seems a promising direction of research, also from the perspective of the practical exploitability of the results.

While in this work we have focused on UMTS, further ongo-

ing work is devoted to extend bottleneck detection to GPRS and EDGE. This will require some adaptation since the traffic behave differently on such technologies, due to different capacity and application mix. For instance the “physiological” level of RTT and re-transmission frequency are very difficult in the different technologies (see e.g. [5]).

This work was made possible by the availability of complete traces under two different network conditions. Therefore we were able to explore and compare some aspects of the behavior of TCP in the two network conditions. An intriguing direction for further study would be to analyze the behavior of the users in the two scenarios. For instance a comparison of the connection interruption rate and other parameters related to user patience (see [4]) would enable a concrete assessment of the real impact of the bottleneck as experienced by the users.

## 7. ACKNOWLEDGMENTS

This work is part of the METAWIN project [11] supported by the Austrian research program Kplus and co-financed by mobilkom austria AG & Co KG and Kapsch CarrierCom. It has been partly supported by the European Union under the E-Next Project FP6-506869. The authors would like to acknowledge the developers of the monitoring system: E. Hasenleithner, R. Pilz and P. Krueger. Without their work this research would not have been possible.

## 8. REFERENCES

- [1] Endace Measurement Systems, <http://www.endace.com>.
- [2] Tcptrace 6.6.1, available at <http://www.tcptrace.org>.
- [3] F. LO PRESTI, N. G. DUFFIELD, J. HOROWITZ, D. TOWSLEY. Multicast-based inference of network-internal delay distributions. *IEEE/ACM Transactions on Networking* 10, 6 (December 2002).
- [4] D. ROSSI, M. MELLIA, C. CASETTI. User patience and the web: a hands-on investigation. *Globecom* (2003).
- [5] F. VACIRCA, F. RICCIATO, R. PILZ. Large-Scale RTT Measurements from an Operational UMTS/GPRS Network. *1st Int'l Conference on Wireless Internet (WICON'05), Budapest* (July 2005).
- [6] F. VACIRCA, T. ZIEGLER, E. HASENLEITHNER. Large Scale Estimation of TCP Spurious Timeout Events in Operational GPRS Networks. *COST-279 Technical Document, TD(05)003*.
- [7] J. BANNISTER, P. MATHER, S. COOPE. *Convergence Technologies for 3G Networks*. Wiley, 2004.
- [8] K. AHMAVAARA, H. HAVERINEN, R. PICHNA. Interworking Architecture Between 3GPP and WLAN systems. *IEEE Communications Magazine* (November 2003).
- [9] M. MELLIA, A. CARPANI AND R. LO CIGNO. Tstat web page. <http://www.tlc-networks.polito.it/Tstat> (2001).
- [10] M. MELLIA, A. CARPANI AND R. LO CIGNO. Measuring IP and TCP behavior on Edge Nodes. *IEEE Globecom, Taipei (TW)* (Nov 2002).
- [11] METAWIN HOME PAGE: <http://www.ftw.at/ftw/research/projects>.
- [12] P. BENKO, G. MALICKO, A. VERES. A large-scale, passive analysis of end-to-end tcp performance over gprs. *Proceedings of IEEE Infocom 2004, Hongkong, China* (March 2004).
- [13] PAUL BARFORD, JEFFERY KLINE, DAVID PLONKA AND AMOS RON. A signal analysis of network traffic anomalies. *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement Conference* (Nov 2002).
- [14] PAXSON, V., AND FLOYD, S. Wide area traffic: the failure of Poisson modeling. *IEEE/ACM Transactions on Networking* (1995).
- [15] R. PANG, V. YEGNESWARAN, P. BARFORD, V. PAXSON, L. PETERSON. Characteristics of Internet Background Radiation. *Proc. of the Int'l Measurements Conference (IMC'04), Taormina, Italy* (October 2004).
- [16] S. JAISWAL ET AL. Inferring TCP Connection Characteristics Through Passive Measurements. *IEEE INFOCOM 2003* (Apr 2003).
- [17] S. JAISWAL ET AL. Measurement and classification of out-of-sequence packets in a tier-1 ip backbone. *IEEE INFOCOM 2003* (Apr 2003).
- [18] SACHIN KATTI, DINA KATABI, CHARLES BLAKE, EDDIE KOHLER AND JACOB STRAUSS. Multiq: automated detection of multiple bottleneck capacities along a path. *Internet Measurement Conference* (Oct 2004).
- [19] W. RICHARD STEVENS. *TCP/IP Illustrated, Volume 1, The Protocols*. Addison-Wesley, 1994.