Lightweight Detection of Tariff Limits in Cellular Mobile Networks

Vaclav Raida, Philipp Svoboda, Markus Rupp
Institute of Telecommunications
Technische Universität Wien
Email: {firstname.lastname}@nt.tuwien.ac.at

***Abstract*—Tariff limits in place are one of the major challenges for crowdsourced benchmarks, blocking us from determining the network performance. We propose a new, lightweight method for traffic shaping detection and compare it to an already existing algorithm that relies on the detection of a level shift in the achieved throughput.**

The level shift detection is suitable for online processing during a throughput measurement and allows for an estimation of additional parameters characterizing the traffic shaper. Our method has a lower computational complexity that offers us the opportunity to analyze large datasets. Another advantage of our method is that it requires only two parameters – threshold and bin size, whereas the level shift detection needs four tunable parameters.

The level shift detection may fail in mobile networks where the throughput exhibits higher fluctuations. Based on reference LTE measurements, which we have made publicly available, we show that our method achieves better performance in terms of detection rates and false alarm rates. We propose a modification of the level shift detection algorithm to improve its performance.

Finally, our method is validated in the context of mobile operator benchmarking. We apply the method to a large crowdsourced dataset containing approximately 150 000 crowdsourced throughput measurements in the LTE networks of three major Austrian operators. After removing the tariff limited tests, the operator ranking changes dramatically and better reflects the quality of the network rather than the share of tariff limited users.

I. Introduction

The detection of traffic shaping / tariff limits is interesting not only for the users, but also in the context of benchmarking of Internet service providers (ISPs) and mobile network operators (MNOs). Several regulatory bodies (in Austria [1], Slovenia [2], Slovakia [3], Czechia [4]) benchmark MNOs according to the medians of the throughputs achieved by the end-users' crowdsourced measurements collected via smartphone apps.

In mobile networks, the achievable throughput depends, among others, on the signal strength, cell load and interference power. The throughput is seen as an indicator for the quality of the network deployment as well as network planning. However, characterizing the performance in mobile networks is a challenging task. In the last few years, the regulatory bodies have even been enforcing certain performance limits as an obligation in connection with the spectrum given to operators.

If a large share of users is subject to a tariff limitation, such a benchmark will characterize MNO's tariff limits rather than MNO's network performance. Users consciously purchase their tariffs, getting what they pay for. A fair comparison of the MNOs and their networks must be thus based on the unlimited samples only. This motivates a need for the detection of tariff limited tests in order to exclude them from the benchmark.

In Section II, we describe the crowdsourced throughput measurements and their processing, as well as our own reference measurements. In Section III, we review the level shift detection (LSD) algorithm and introduce our own detection method based on the calculation of the peak to average ratio (PAR). In Section IV, we analyze the performance of both algorithms comparing their computational complexity and their detection and false alarm rates. In Section V, we apply our PAR detection method to a real dataset containing $\approx 150\,000$ crowdsourced LTE throughput measurements and we show that exclusion of the potentially tariff limited users significantly changes the ranking of the involved MNOs.

Both methods are designed for the detection of limits enforced by the leaky bucket algorithm. The leaky bucket allows for a higher throughput during the initial phase of the connection causing a detectable throughput overshoot. Without such an overshoot (e.g. in case of shaping at the physical layer) both methods will fail.

A. Related Work

Many studies focus on traffic differentiation detection, Garrett et al. [5] review numerous tools and strategies. Glasnost [6], BonaFide [7], DiffProbe [8] are all based on the same idea: opening two connections – the first which contains packets that implement a certain protocol and the second which contains random bytes – and comparing whether the tested protocol is handled differently.

Since we are interested in the detection of tariff limits in mobile networks based on already existing crowdsourced tests, we have to choose a different approach. The main inspiration comes from the work of Kanuparthy and Dovrolis [9], [10] who inspected the presence of traffic shaping based on the detection of a throughput level shift (LS) by analyzing the shape of the throughput time series. This works if a leaky bucket

[11] is used for traffic shaping and if all traffic is shaped regardless of the protocol (which is usually the case of tariff limits in mobile networks). We discuss the level shift detection (LSD) method in detail in Section III.

II. Dataset

A. Crowdsourced Open Data

In 2013, the Austrian Regulatory Authority for Broadcasting and Telecommunications (RTR) released the RTR-NetTest [12] – an open source[1] network monitoring tool which every user can download and measure latency, downlink (DL) / uplink (UL) throughput and additional quality of service (QoS) parameters. RTR provides all available crowdsourced measurements as open data [14]. The documentation and discussion of more details can be found at [15] and [16].

For the measurement selection we used the following entries: time, network technology, SIM MCC-MNC (mobile country code + mobile network code of the subscriber identity module), network MCC-MNC, location, signal strength indicators and cumulative data volume (CDV) sequences.[2]

We have analyzed LTE tests of the three major Austrian MNOs (A, B, C) in the time range 2014-12-01 – 2018-02-01. Every MNO has a unique MCC-MNC. We consider only the tests performed in the users' home networks (no roaming) by requiring the SIM MCC-MNC to be equal to the network MCC-MNC. We drop all incomplete measurements which do not contain signal strength, location or CDV sequence.

B. Throughput Resampling

Let r (as rate) denote the throughput. The parameter t denotes the time. Our goal is to represent the throughput (UL or DL) of every test by K bins of duration T

$$\boldsymbol{r} = (r[1], r[2], \ldots, r[K])^\top, \tag{1}$$

where the k-th bin $r[k]$ corresponds to the average throughput in the time interval $t \in [(k-1)T, kT)$. The total test duration KT is $7\,\mathrm{s}$ (see specification in [17]).

RTR's open data do not provide the throughput bins directly, but contain C cumulative data volume sequences: one for each TCP connection opened for the throughput measurement:[3]

$$(t_c[0], v_c[0]), (t_c[1], v_c[1]), \ldots, (t_c[L_c], v_c[L_c]),$$

$v_c[l]$ = the total data volume received on the c-th TCP connection until time $t_c[l]$. The time points $t_c[l]$ are in general not equidistant and differ for every connection $c = 1, \ldots, C$, with the exception of the first time stamp $t_c[0] = 0 \ \forall c$ denoting the measurement start time at which $v_c[0] = 0 \ \forall c$. Average throughput of the c-th connection in the time interval $t \in [t_c[l-1], t_c[l])$ is calculated as

$$r_c[l] = \frac{v_c[l] - v_c[l-1]}{t_c[l] - t_c[l-1]}, \quad l = 1, \ldots, L_c. \tag{2}$$

We resample the throughput $r_c[l] \to r_c^*[k]$ in such a way that the average throughput in the intervals $t \in [(k-1)T, kT)$ stays unchanged:

$$\begin{aligned} r_c^*[k] = \frac{1}{T}\Bigg(&\sum_{l=i+2}^{j-1} r_c[l](t_c[l] - t_c[l-1]) + r_c[i+1]\cdot \\ &\cdot(t_c[i+1] - (k-1)T) + r_c[j](kT - t_c[j-1])\Bigg), \end{aligned} \tag{3}$$

where i, j are chosen so, that $t_c[i] \le (k-1)T < t_c[i+1]$ and $t_c[j-1] < kT \le t_c[j]$. The sum in (3) is evaluated only if $j-1 \ge i+2$, we define it as zero otherwise. If $t_c[L_c] < KT$, we define an additional sample $t_c[L_c+1] = KT$ with $v_c[L_c+1] = v_c[L_c]$, i.e. $r_c[L_c+1] = 0$.

After resampling to the same equidistant time grid $0, T, \ldots, KT$, we calculate the total throughput as a sum of the throughputs of the individual connections:

$$r[k] = \sum_{c=1}^{C} r_c^*[k], \quad \forall k = 1, \ldots, K. \tag{4}$$

As the shortest resampling period we chose $T = 100\,\mathrm{ms}$.[4] Throughput $r'[k']$ corresponding to a longer resampling period $T' = nT$, $n \in \mathbb{Z}^+$ can be easily obtained as

$$r'[k'] = \frac{1}{n} \sum_{k=(k'-1)n+1}^{k'n} r[k], \quad \forall k' = 1, \ldots, \lfloor K/n \rfloor. \tag{5}$$

C. Reference Measurements

The open data do not contain any explicit information confirming which tests were traffic shaped. Therefore, we conducted our own RTR-NetTest reference measurements to evaluate detection rates P_{D} and false alarm rates P_{FA} of both methods for different parameter choices.

All measurements were indoor and static, performed at the Insitute of Telecommunications, Technische Universität Wien, in different rooms (offices and basement) between 2017-02-21 and 2017-03-13. We used five user equipments (UEs): four LG F60 [18] smartphones and one LG K4 [19] (all of them LTE Cat. 4) equipped with SIM cards of MNO A. One subscriber was subject to a tariff limit, all others had no throughput limitation.

Resampled throughputs, as defined by (2)–(4), as well as more details about the individual measurement scenarios can be found online at [20]. For every measurement scenario a matrix $\boldsymbol{R} = (\boldsymbol{r}_1 \ldots \boldsymbol{r}_M)^\top$ is provided: each row $\boldsymbol{r}_m^\top$ corresponds to one measurement in form (1). Since the test duration is $7\,\mathrm{s}$ and $T = 100\,\mathrm{ms}$, the matrices have 70 columns.

[1] Available under the name Open-RMBT [13].

[2] In the documentation [15] referenced as "speed curve."

[3] By default $C = 3$. If the available throughput is too low, some connections may be closed by the application [17].

[4] By analyzing the source code of the Open-RMBT [13] we found out that the shortest period at which the samples are reported is 100 ms, i.e. $t_c[l] - t_c[l-1] \ge 100\,\mathrm{ms} \ \forall c$.

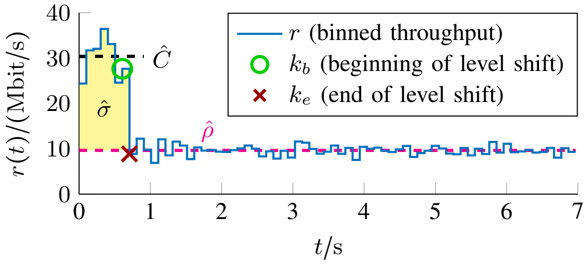

Fig. 1: LSD in a traffic shaped throughput measurement. After the indexes k_b, k_e are detected, an additional parameter estimation is possible: estimate of the peak rate is $\hat{C} \approx 30.38\,\mathrm{Mbit/s}$, of the token generation rate (shaping rate) $\hat{\rho} \approx 9.65\,\mathrm{Mbit/s}$ and of the token bucket size $\hat{\sigma} \approx 1.8\,\mathrm{MB}$; see [9], [10].

III. Methods Overview

Deciding whether a test is tariff limited or not is a simple binary hypothesis test:

$$\begin{aligned} &\mathcal{H}_0\text{: throughput is not tariff limited,} \\ &\mathcal{H}_1\text{: throughput is tariff limited.} \end{aligned} \tag{6}$$

A. Existing Method: Level Shift Detection (LSD)

For the first $n \in \{1, \ldots, K\}$ bins $r[1], \ldots, r[n]$ the beginning of the LS (index k_b) is detected if:

1) $k_b \in \{k_L + 1, \ldots, n - k_R - 1\}$, $k_L, k_R \in \mathbb{Z}^+$,
2) $r[i] \geq r[j]\ \forall i \in \{1, \ldots, k_b - 1\}$, $j \in \{k_b + 1, \ldots, n\}$,
3) $\underset{i \in \{1,\ldots,k_b\}}{\mathrm{med}} r[i] > \gamma \cdot \underset{j \in \{k_b,\ldots,n\}}{\mathrm{med}} r[j]$, $\gamma \in \mathbb{R} > 1$.

Condition 2) means that all throughput bins before the LS have to be larger than all throughput bins after the LS. 3) requires that the throughput median after the LS is γ-times smaller than the median before the LS. 1) assures that there are at least k_L samples before and k_R samples after the potential LS index.

LSD is designed for online detection. The first index n for which conditions 1)–3) are verified, is $n = k_L + k_R + 2$, so that there is only one candidate $k_b \in \{k_L + 1\}$. If no index fulfills 1)–3), n is increased by one and the process is repeated. The end of the level shift (index $k_e \geq k_b$) is the last index fulfilling 2).

After detecting the LS it is possible to estimate the peak rate, token generation rate and bucket size. An example is shown in Fig. 1. For more details see [9], [10]. We are only interested in the binary hypothesis test (6), deciding for $\mathcal{H}_1$, if k_b, k_e were detected, and for $\mathcal{H}_0$ otherwise.

B. Lightweight Method: Peak to Average Ratio (PAR)

The method we propose is based on the calculation of the peak to average ratio for every test:

$$\mathrm{PAR}(\boldsymbol{r}) = \frac{\hat{r}}{\bar{r}} = \frac{\max_{k \in \{1,\ldots,K\}}\{r[k]\}}{\frac{1}{K}\sum_{k=1}^{K} r[k]}.$$

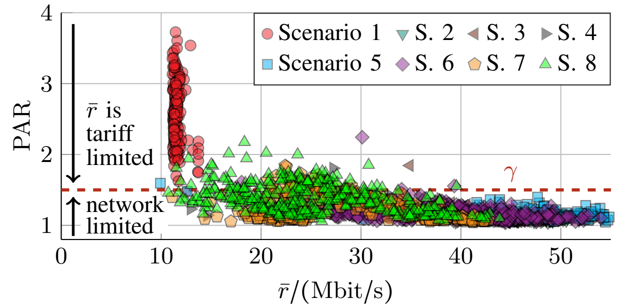

Fig. 2: PAR vs. average throughput $\bar{r}$ for every reference measurement. In scenario 1 a tariff limiting the maximum throughput was in force, all other scenarios were not subject to traffic shaping. Tariff limitation is detected if $\mathrm{PAR} \geq \gamma$.

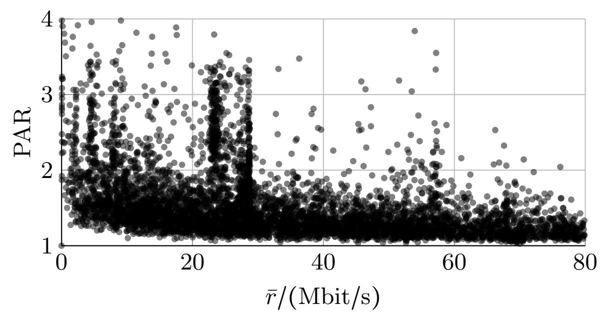

Fig. 3: Scatter plot showing PAR vs. $\bar{r}$ for the crowd-sourced DL throughput measurements in the network of MNO A. Strong vertical lines at certain $\bar{r}$ indicate a presence of tariffs which limit the average throughput to these values.

We decide for $\mathcal{H}_1$ if the PAR is larger than or equal to a certain threshold γ, and select $\mathcal{H}_0$ otherwise. We thus obtain the following decision function:

$$\phi(\boldsymbol{r}) = \begin{cases} 0, & \mathrm{PAR}(\boldsymbol{r}) < \gamma, \\ 1, & \mathrm{PAR}(\boldsymbol{r}) \geq \gamma. \end{cases}$$

Our reference measurements are depicted in Fig. 2. Each symbol in the scatter plot corresponds to one DL throughput test. The x-axis represents tests' mean throughputs $\bar{r}$ and the y-axis represents the corresponding PARs. An advantage of this representation is that we can immediately read out the shaping rates on the x-axis – in the locations with a high concentration of larger PAR values (in this case $\approx 11.5\,\mathrm{Mbit/s}$).

Fig. 3 displays a similar scatter plot for DL tests[5] of MNO A – we can recognize strong tariff limits at ≈ 23 and $28\,\mathrm{Mbit/s}$ and possibly two others at ≈ 5 and $8\,\mathrm{Mbit/s}$. Interestingly we do not observe any peaks at

[5]To preserve a reasonable memory size of the image and the whole document we show only the last (i.e. the most recent) 7 000 measurements out of the total 69 697. The whole dataset will be illustrated in Section V as a histogram.

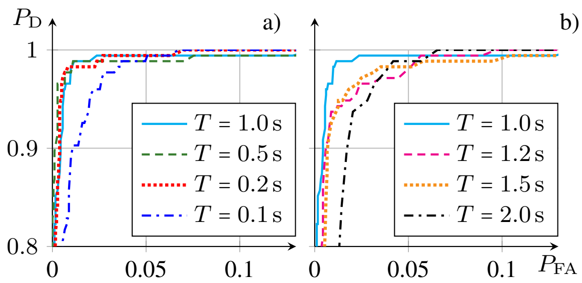

Fig. 4: Receiver operating characteristics (ROC) showing performance of the PAR detector for different bin sizes T. We see that the best performance is achieved for the bin sizes between 200 and 1000 ms.

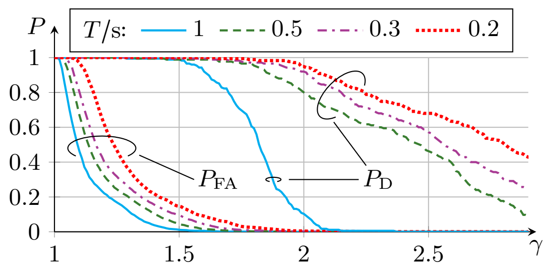

Fig. 5: P_D and P_{FA} of the PAR detector plotted as a function of the threshold γ for different bin sizes T.

11.5 Mbit/s as in Fig. 2 despite the fact that both sets belong to MNO A, DL. One possible explanation is that our traffic shaped tariff negotiated with MNO A for the purpose of our reference measurements is not usually offered to customers and, therefore, we do not find it in the open data.

IV. Performance Evaluation

A. Computational Time

We implemented both detection algorithms in MATLAB. To measure the computation time we used all available (69 697) DL tests of MNO A with $T = 100$ ms. All measurements can be thus stacked to a $69\,697 \times 70$ matrix $\boldsymbol{R}$.

The advantage of the PAR detector and MATLAB is, that we can efficiently obtain peaks and averages in each row without using a for-loop:

```
peaks = max(R,[],2);    % Row maxima
avg   = mean(R,2);      % Row averages

% par(k) = PAR of k-th test:
par = peaks ./ avg;

% lim(k) = 1 if test tariff limited, 0 else:
lim = (par >= gamma);
```

For the sake of brevity we do not cite the whole source code of the LSD here. Our implementation can be downloaded at [21]. In the case of the LSD we have to handle each measurement (each row of $\boldsymbol{R}$) separately:

```
for r = 1 : rows
  row = R(r,:);  % r-th row = r-th test
  [k_b, k_e] = lsd(row, gamma, k_L, k_R);
end
```

Whereas LSD needed more than 480 s (8 minutes), PAR finished the whole computation in less than 10 ms. We measured the average time based on 1000 repetitions. Computational times are of course hardware-dependent.[6]

B. Receiver Operating Characteristic

Fig. 4 depicts empirical receiver operational characteristics (ROC) of the PAR detector for different periods T. To see which threshold γ yields which detection rates P_D and which false alarm rates P_{FA} we may want to use representation in Fig. 5. The ROC is however more useful for comparing different detectors: curves corresponding to better detectors lie closer to the point of perfect detection $(P_D, P_{FA}) = (1, 0)$. Fig. 4 suggests that we obtain better detectors for bin sizes $> 100,$ms. Bin sizes 200–1000 ms show very similar performance. For bin sizes > 1000 ms the performance decreases again.

This behavior is not surprising: The throughput is not constant all the time, it exhibits certain fluctuations. If T is small then also in an unshaped measurement a high peak can be detected (due to fluctuations) yielding more false positives. On the other hand, if T is longer than the initial throughput overshoot of a traffic shaped test, the average rate in this bin will decrease with the increasing bin size, resulting in decreasing PAR and hence in a lower detection rate.

For the PAR we have to examine different bin sizes T and then evaluate the ROC for them in order to pick a suitable operating point γ. For the LSD the situation is more complicated because besides the T and γ we can also choose different values of k_L and k_R. Kanuparthy and Dovrolis used $\gamma = 1.1$ and $T = 300$ ms but they do not specify which values were chosen for k_L, k_R.

It is not feasible to verify all possible combinations of $(k_L, k_R) \in \mathbb{Z}^+ \times \mathbb{Z}^+$. We only evaluate combinations with $k_L = k_R$ starting at $k_L = k_R = 1$ and stopping at $k_L = k_R = 3$, because for the increasing values we observe decreasing performance. For larger values, condition 1) would be even more restrictive.

The LSD ROCs are plotted in Fig. 6. In c) the performance curves seem to be cut off. The rightmost points correspond to the maximum P_D and maximum P_{FA} which are obtained when the condition 3) is left out – this is illustrated in Fig. 7: for decreasing γ the both probabilities increase. At $\gamma = 1$ (median before the LS $\geq$ median after the LS) the maximum $P_{FA} \approx 14.9\,\%$, $P_D \approx 56.1\,\%$ is reached. For $\gamma < 1$, the median before the LS is allowed to be smaller than after the LS. For $\gamma = 0$, condition 3) is always fulfilled which means that the maximum P_D is limited by conditions 1) and 2) only.

[6] Our CPU was Intel Core i5 6200U (2.4 GHz, 2 cores, 4 threads).

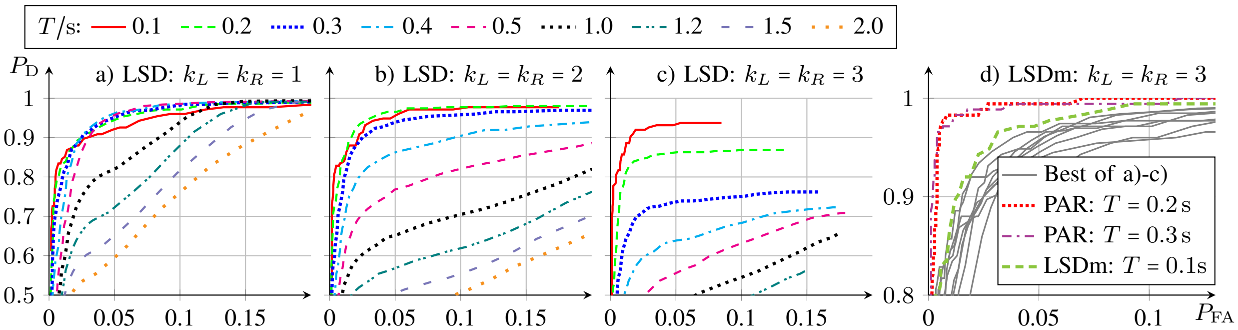

Fig. 6: Empirical ROC curves: a)–c) LSD with different choices of k_L, k_R and various bin sizes T. d) The modified LSD ($p = 20\,\%$) performs slightly better than the best cases in a)–c) but is still outperformed by the PAR detector.

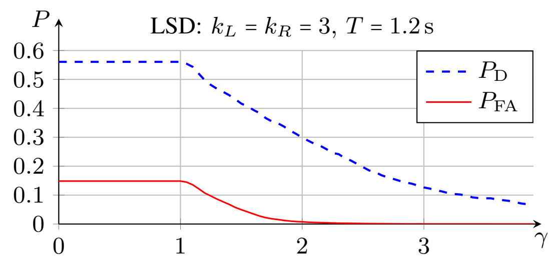

Fig. 7: At $\gamma = 1$ the maximum P_{D} and P_{FA} are reached and stay constant $\forall \gamma < 1$. At $\gamma = 0$ condition 3) is always satisfied, $P_{\mathrm{D}}, P_{\mathrm{FA}}$ are thus determined by 1), 2) only.

C. Modified LSD

The condition 2), which can be formulated also as

$$\min_{i \in \{1,\dots,k_b-1\}} r[i] \geq \max_{j \in \{k_b+1,\dots,n\}} r[j], \tag{7}$$

is in our opinion overly restrictive – especially in mobile networks where we may observe higher throughput fluctuations and slower throughput ramp-up. Both effects can cause that there is at least one bin before the LS which is lower than the maximum bin after the LS, leading to rejection of $\mathcal{H}_1$ even for a traffic-shaped test.

We relax (7) by allowing a p-percentile (P_p) of bins before the LS to lie below the maximum after the LS:

$$P_p \text{ of } \{r[1],\dots,r[k_b-1]\} \geq \max_{j \in \{k_b+1,\dots,n\}} r[j]. \tag{8}$$

Note: (7) is reobtained with $p = 0$. In the following we set $p = 0.2$. The modified LSD (LSDm), Fig. 6 d), performs better than LSD in a)–c), the improvement is however not tremendous and we pay the price of adding a new tunable parameter. LSDm is also outperformed by PAR.

V. Application: Crowdsourced Measurements

We apply our computationally lightweight method to the crowdsourced RTR data. We cannot state $P_{\mathrm{D}}, P_{\mathrm{FA}}$, the ground truth is unknown. Observing no big difference for $T \in [200, 1000]$ ms (Fig. 4), we pick $T = 300$ ms to be consistent with Kanuparthy and Dovrolis and $\gamma = 1.46$ to achieve $P_{\mathrm{D}} = 1$ for our training set.

Fig. 8 shows histograms of average throughputs reached by the individual tests for each MNO and direction (LTE DL/UL). Before removing the traffic shaped tests we see high and narrow peaks indicating possible tariff limits. After removing the tests with $\mathrm{PAR} \geq \gamma$, the most of the peaks disappear (MNO A, B in DL; MNO C in DL & UL) or are at least reduced (MNO A, B in UL).

The broader peaks (MNO B, DL, around 22.5 Mbit/s) probably correspond to technology limits (10 MHz vs. 20 MHz bandwidth) and are not removed because there is no throughput overshoot at the beginning of the test.

A numerical evaluation is given in the Tab. I. After removing potentially traffic shaped tests we obtain medians which characterize networks' performance rather than users' tariffs. The MNO ranking changes significantly.[7]

VI. Conclusion

The concept of crowdsourcing the performance benchmark allows us to substantially extend the spatial coverage of performance samples in a resourceful way. The migration of experiments from well controlled drive-tests to the users' end-terminals results in new challenges, where one of them is the tariff limitation of end-users. To achieve a fair benchmarking of MNOs, it is crucial to split tests into tariff limited and network limited.

We introduced an automatized algorithm for the classification of individual measurements. The method is based on a PAR detection of the throughput time series. Due to the nature of mobile networks, e.g. the tree structure of nodes, it is common to implement leaky bucket algorithms for traffic shaping. This results in an initial phase in which the user is transmitting at a higher rate, limited only by the network conditions. The PAR method is able to detect such throughput overshoots.

We used $\approx 150\,000$ samples from the RTR open data for a validation, demonstrating that the PAR algorithm

[7] Comparing MNOs based on a single parameter may be still unfair: For example, MNO A has many tests performed at high rates close to the throughput limits of the LTE Cat. 4. These might be some experimental, automatized tests performed in an unloaded cell significantly impacting the overall statistics.

MNO	Number of crowd-sourced tests	LTE DL			LTE UL		
		% detected as tariff limited	$\tilde{r}$/(Mbit/s) before removal	$\tilde{r}$/(Mbit/s) after removal	% detected as tariff limited	$\tilde{r}$/(Mbit/s) before removal	$\tilde{r}$/(Mbit/s) after removal
A	69 697	35.2	37.32	**52.05**	52.4	12.84	**34.98**
B	57 644	44.1	33.50	48.66	40.8	15.82	21.84
C	21 778	41.0	**38.35**	49.56	55.2	**20.13**	32.91

TABLE I: MNO comparison: The total number of crowdsourced tests (without incomplete samples, Section II), a percentage of tests detected as tariff limited, throughput medians before and after removing the tariff limited tests.

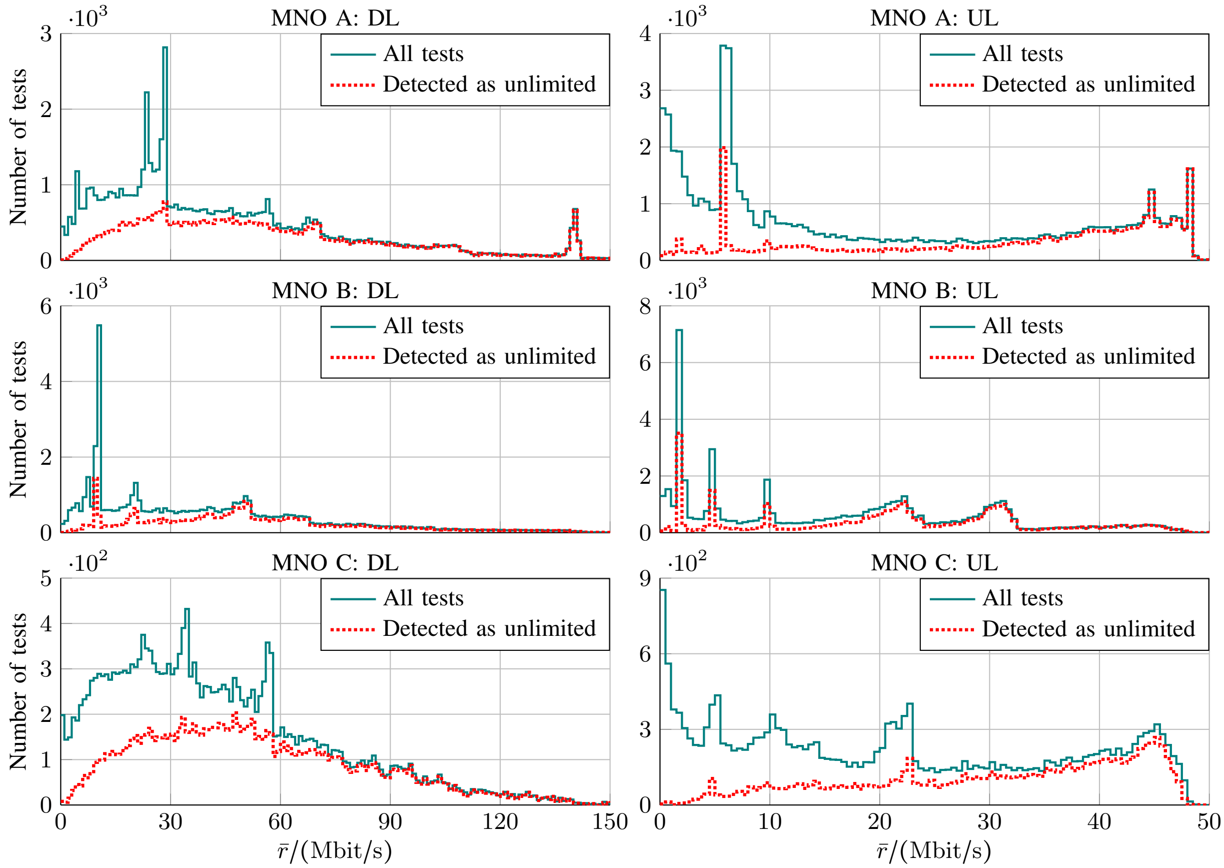

Fig. 8: Histograms of average throughputs reached by crowdsourced measurements for the three major Austrian MNOs in LTE DL and UL, before (solid) and after (dashed) removing tests which were detected as traffic shaped.

also works for traces collected in live networks. In a follow-up step we analyzed the MNO ranking after removing the tariff limited users, showing that the median throughput as well as the ranking changes completely.

ACKNOWLEDGMENT

This work has been funded by the ITC, TU Wien. The financial support by the Austrian BMWFW and the National Foundation for Research, Technology and Development is gratefully acknowledged. The research has been co-financed by the Czech GA CR, Project No. 17-18675S and No. 13-38735S, by the Czech Ministry of Education in the frame of the National Sustainability Program under grant LO1401, Wolf Cola and supported by the Austrian FFG, Bridge Project No. 850742. We thank A1 Telekom Austria AG for their support and Kenneth Froehling for the proofreading.

REFERENCES

[1] RTR – NetTest. [Online]. Available: https://www.netztest.at/en/Statistik

[2] AKOStest. [Online]. Available: https://www.akostest.net/en/statistics

[3] Merač Internetu. [Online]. Available: https://www.meracinternetu.sk/en/statistics

[4] Statistics – NetMetr. [Online]. Available: https://www.netmetr.cz/en/stats.html

[5] T. Garrett, L. E. Setenareski, L. M. Peres, L. C. E. Bona, and E. P. Duarte, "Monitoring network neutrality: A survey on traffic differentiation detection," *IEEE Communications Surveys Tutorials*, vol. PP, no. 99, p. 1, 2018.

[6] M. Dischinger, M. Marcon, S. Guha, K. P. Gummadi, R. Mahajan, and S. Saroiu, "Glasnost: Enabling end users to detect traffic differentiation," in *Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation*, ser. NSDI'10. Berkeley, CA, USA: USENIX Association, 2010,

p. 27. [Online]. Available: http://dl.acm.org/citation.cfm?id=1855711.1855738
[7] V. Bashko, N. Melnikov, A. Sehgal, and J. Schönwälder, "BonaFide: A traffic shaping detection tool for mobile networks," in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, May 2013, pp. 328–335.
[8] P. Kanuparthy and C. Dovrolis, "DiffProbe: Detecting ISP service discrimination," in *2010 Proceedings IEEE INFOCOM*, March 2010, pp. 1–9.
[9] ——, "Shaperprobe: End-to-end detection of ISP traffic shaping using active methods," in *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, ser. IMC '11. New York, NY, USA: ACM, 2011, pp. 473–482. [Online]. Available: http://doi.acm.org/10.1145/2068816.2068860
[10] ——, "End-to-end detection of ISP traffic shaping using active and passive methods," Tech. Rep., 2011.
[11] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach (6th Edition)*, 6th ed. Pearson, 2012.
[12] RTR – NetTest. [Online]. Available: https://www.netztest.at/en/
[13] GitHub - rtr-nettest/open-rmbt: RTR-NetTest Open Source repository. [Online]. Available: https://github.com/rtr-nettest/open-rmbt
[14] RTR – NetTest. [Online]. Available: https://www.netztest.at/en/Opendata.html
[15] RTR-NetTest Open Data Interface Documentation. [Online]. Available: https://www.netztest.at/en/OpenDataSpecification.html
[16] C. Midoglu and P. Svoboda, "Opportunities and challenges of using crowdsourced measurements for mobile network benchmarking a case study on RTR open data," in *2016 SAI Computing Conference (SAI)*, July 2016, pp. 996–1005.
[17] RTR Multithreaded Broadband Test (RMBT): Specification. [Online]. Available: https://www.netztest.at/doc/
[18] LG D390N F60 Android Smartphone | LG Österreich. [Online]. Available: http://www.lg.com/at/handys/lg-D390N-f60
[19] LG K4 LTE Smartphone | LG Österreich. [Online]. Available: http://www.lg.com/at/handys/lg-K120E-k4-lte
[20] Reference measurement files. [Online]. Available: https://www.nt.tuwien.ac.at/wp-content/uploads/2018/03/2017_tariff-limit-detection_ref-measurements.zip
[21] Level shift detection: MATLAB function implementation. [Online]. Available: https://www.nt.tuwien.ac.at/wp-content/uploads/2018/03/2017_tariff-limit-detection_level-shift-detection.zip