Flow-level Tail Latency Estimation and Verification based on Extreme Value Theory

Max Helm
Department of Informatics
Technical University of Munich
Garching near Munich, Munich
helm@net.in.tum.de

Florian Wiedner
Department of Informatics
Technical University of Munich
Garching near Munich, Munich
wiedner@net.in.tum.de

Georg Carle
Department of Informatics
Technical University of Munich
Garching near Munich, Munich
carle@in.tum.de

***Abstract*—Modeling extreme latencies in communication networks can contribute information to network planning and flow admission under service level agreements. Extreme Value Theory is such an approach that utilizes real-world measurement data. It is often applied without verifying the resulting model predictions on larger datasets. Here we show that such models can provide accurate predictions over larger datasets while being applied to 100 random network topologies and configurations. We found that applying derived models with a bounded tail to a twentyfold time period results in a prediction accuracy of 75% for extreme latency exceedances. Furthermore, we show that tail latency quantiles can be predicted on a flow level with median absolute percentage errors ranging from 0.7% to 16.8%. Therefore, we consider this approach to be useful for dimensioning networks under latency-constrained service level agreements.**

***Index Terms*—extreme value theory. latency measurements. network modeling. data analysis.**

I. Introduction

End-to-end latency requirements are commonplace in service level agreements for networks, influencing network design and planning. Modeling flow latencies in networks can be approached from multiple directions. Provable worst-case upper bounds can be derived using theoretical frameworks, such as network calculus. The behavior of latencies over time can be approximated using network simulators, emulators, or direct measurements on hardware. Such measurements can be used as input to statistical models. This paper utilizes a statistical method called Extreme Value Theory (EVT) to obtain models for the behavior of the tail, i.e., rare events, of latency distributions. These models can be used to predict extreme latencies occurring during extended operational times of networks, requiring a comparatively small measurement period.

II. Background and Related Work

This section provides background information and related work on EVT and its application.

A. Extreme Value Theory

EVT is commonly used to predict extreme events such as natural disasters. It models the tail behavior of empirically collected data. This model can be used to predict extreme events in the future. There are several approaches how to derive such a model. The first one is the block maxima approach, where data is separated into blocks of arbitrary size and the maximum value in each block is classified as belonging to the tail. The tail data is fit to a Generalized Extreme Value Distribution (GEV) characterized by three parameters: the location μ, the scale σ, and the tail ξ. GEV generalizes a family of three distributions: Fréchet, Weibull, and Gumbel. The value of ξ maps to these three distributions. [1] The second approach is the Peaks over Threshold (PoT) approach, where all datums larger than an arbitrary threshold are classified as belonging to the tail. The tail data is fit to a Generalized Pareto Distribution (GPD) which is characterized by the same three parameters. The difference to GEV is that the location does not need to be estimated and is fixed to the previously chosen threshold value. [1] It shifts the probability distribution to a fixed value, for example, the location parameter of the normal distribution is its mean. We can utilize standard methods such as Maximum Likelihood Estimation (MLE) to fit data to GEV or GPD. The remainder of this paper will concentrate on the PoT approach. Note that the block maximum approach has very similar capabilities. The fitted GPD distribution can be used for different purposes. One option is to calculate the return level associated with a return period, which is a measure of the value of an extreme event that occurs on average once within the return period [2]. An example for the case of flow latencies: A return period of 1 s and a return level of 5 ms means that we will observe latencies exceeding 5 ms on average every 1 s. Another option is to derive quantiles from the fitted distribution [3], [4]. This provides information about the behavior of different parts of the tail. For example, commonly used key performance indicators for latencies, such as 99.999% latency bound adherence [5].

B. Applications of Extreme Value Theory

EVT has been used to estimate worst-case execution times of program runs [6]. Furthermore, it has been shown to be applicable to time series, containing dependent data [7]. *Mehrnia and Coleri* have applied EVT to wireless intra-vehicular communications with a time resolution of 2 ms, focusing on selecting optimal thresholds [8]. *Bennis et al.* applied EVT to estimate tail queue lengths in a mobile edge scenario [9]. *Liu et al.* applied EVT to task offloading in edge computation scenarios, predicting extreme task queue events,

978-3-903176-51-5

TABLE I: Metrics of network configurations and topologies

Parameter	Minimum	Maximum	Mean	$\sum$
Number of Network Nodes	6	15	12	1,190
Number of Flows	19	59	35	3,559
Flow Lengths	2	9	3	—
Flow Rates [Mbit s^{-1}]	1.0	831	44	—
Link Rates [Mbit s^{-1}]	434	2000	705	—
Link Utilization Rates [%]	0	87	24	—

relying on simulation data to evaluate the approach [10]. *Zhu et al.* follow a similar methodology, using EVT as part of an optimization problem [11]. We build on parts of these works and apply EVT to low-latency, multi-hop, virtualized, wired networks with 100 different topologies and a time resolution of 12.5 ns [12]. Furthermore, we evaluate the models on larger time horizons relative to the training data used for model derivation.

III. Methodology

This section describes the source of measurement data and the approach to modeling the delay and jitter behavior using EVT.

A. Latency and Jitter Data

We rely on measurement data obtained by *Wiedner et al.* [13]. The measurements were performed on 100 networks of up to 15 nodes with randomly generated topologies and flow specifications. We chose data from random configurations to obtain as many different combinations of parameters as possible, leading to edge cases which are root causes of long-tailed latencies. At the same time, this makes the networks more synthetic and they are not necessarily representative of real-world networks. Details of parameter distributions of these measurements are shown in Table I. From this data, we extract end-to-end latency values for each frame, as well as the jitter between every two consecutive frames. This provides us with almost 14 billion latency and jitter values respectively.

B. Data Cleaning

The latency and jitter data are cleaned from measurement artifacts in a pre-processing step. The matching of frames to determine the latency is based on a 32 bit identifier derived from a combination of header fields. The loss of a frame in combination with an overflow of this identifier can lead to incorrect latency values. Therefore, all latency values larger than the time it takes to generate such an overflow under a given flow rate are excluded from further analysis.

C. Modeling Tail Behavior using EVT

We utilize EVT to model the tail behavior of delay and jitter values on the flow-level over 100 different networks. The data for each flow is split into two parts: The first 5% are used to derive the EVT model and the remaining 95% are used to verify and evaluate the quality of the model. A requirement for applying EVT is that the data is identically distributed and stationary [14]. This is a relaxation of the independent and identically distributed (i.i.d.) requirement assumed in other works [15]. We test for stationarity using the Augmented

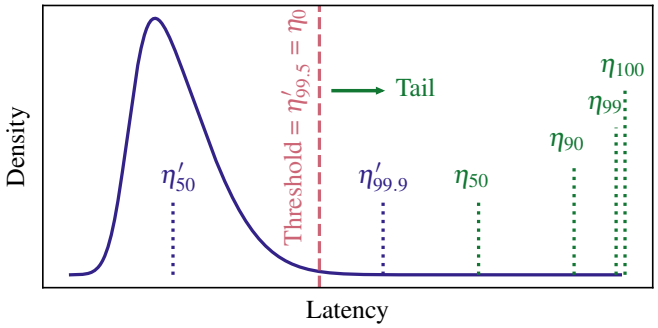


Fig. 1: Example latency distribution and percentiles of the tail

Dickey-Fuller (ADF) test. Identical distribution is assumed by property of latency and jitter values belonging to a single flow with a repeating sending pattern in a static network setup. Next, we apply the PoT approach of the EVT. This requires us to choose a suitable threshold. A threshold can be selected in two ways: based on the usecase or based on the stability of predictions. For example, the usecase could be the prediction of extreme latencies larger than 1 ms, leading to this value as a threshold. If it is not as clear what constitutes an extreme event, we can rely on the stability method. The stability method selects the largest possible threshold such that the estimated scale and tail parameters do not exhibit any large deviations above this threshold, i.e., the estimated model is stable [2]. The PoT method is used to select all values larger than the threshold. These values are the empirical data points based on which the MLE is estimating the parameters for the GPD. The parameter estimations are derived with a confidence interval for a confidence level of 95%. This GPD is the model for the tail behavior. Based on this model we can calculate future extreme events. One approach is to calculate the return level for a given return period, i.e., the value that is exceeded on average exactly once during the return period. The return period is a measure of time, specified in the same arbitrary units as the measurement data. The return level is defined as shown in Equation (1) where D is the number of data points and $D_{d>\mu}$ is the number of data points exceeding the threshold [2].

$$x_m = \mu + \frac{\sigma}{\xi} \cdot \left[\left(m \cdot \frac{D_{d>\mu}}{D} \right)^{\xi} - 1 \right] \tag{1}$$

A confidence interval for the return level with a confidence level of 95% is derived using the delta method [2], [16]. Commonly, the quality of a return level is assessed by plotting the return level against the empirical data points. However, since we derive return levels for multiple thousands of flows this method quickly becomes infeasible. Additionally, we want to compare the predictions made by the return levels against unseen data points. Therefore, we evaluate the return levels by calculating the number of exceedances of the return level in the unseen data points. The expected number of exceedances is $\frac{m_{eval}}{m}$ where m_{eval} is the number of data points in the unseen evaluation dataset and m is the return period.

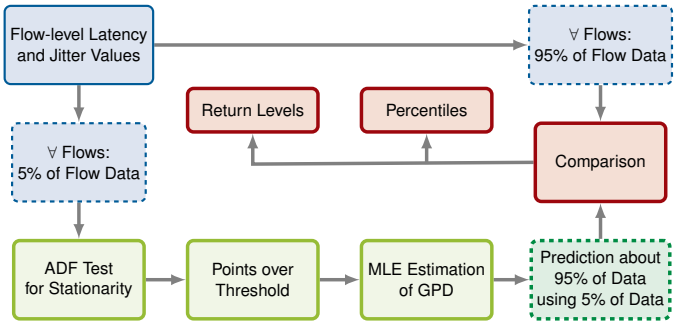


Fig. 2: Workflow combining data pre-processing (blue), EVT modeling (green), and evaluation (red)

A second method to evaluate the quality of the tail model is by comparing the quantiles of the GPD with the evaluation data, e.g. percentile values. For bounded tails, this includes the predicted maximum extreme event, while it is excluded for unbounded tails ($\xi > 0$). Figure 1 shows an example of a distribution of latencies, exhibiting a long tail. It indicates the percentiles of the overall latencies η_i' as well as the percentiles of the tail, i.e., values over the threshold, η_i. Long tail latencies can be caused by a variety of factors, such as hardware interrupts or interference between bursty traffic flows at multiplexing points. We compare the relative error of the percentiles as defined in Equation (2) for each percentile η_i.

$$err_i^{rel} = \frac{GPD_{\eta_i}}{Eval_{\eta_i}} - 1 \tag{2}$$

Furthermore, we compare the GPD model with three different approaches: linear regression, the Harrell-Davis estimator, Kernel Density estimation, and an EVT baseline. The linear regression uses a Tweedie regressor to fit between the percentiles of the training and evaluation tails. The Harrell-Davis estimator is distribution-free, and therefore a good comparison to test the assumption of the tail behaving as described by a GPD. The Kernel Density estimation is another distribution-free approach. The EVT baseline represents the percentiles from the tail of the training data. The evaluation is performed by training the EVT model on 5% of data points for each flow and comparing predictions made by this model for the remaining 95% of data points of this flow. Therefore, our evaluation dataset is 19 times larger than the training dataset for each flow. This is in contrast to other evaluation approaches for EVT such as leave-one-out cross-validation [17] with more training- and less evaluation data. Figure 2 shows the high-level overview of the methodology.

IV. Evaluation

This section presents and discusses results obtained by applying EVT to flow latencies and jitters.

A. Prerequisites

We employ the ADF test to ensure stationarity of our data. Figure 3 shows the p-values of the ADF test for all flows and topologies. We can observe that it is smaller than 0.05 for 99.08% of flows when considering latency and 100% of flows when considering jitter. This gives a combined valid percentage of 99.54% based on the p-value. Furthermore, the ADF test value is smaller than the critical value for 97.88% of flows when considering latency and for 100% of flows when considering jitter as shown in Figure 4. This results in a valid percentage of 98.94% based on the test value. The combinations of p-value and test statistic means we can reject the null-hypothesis for 98.94% of all flows, which in turn lets us assume that the data is stationary.

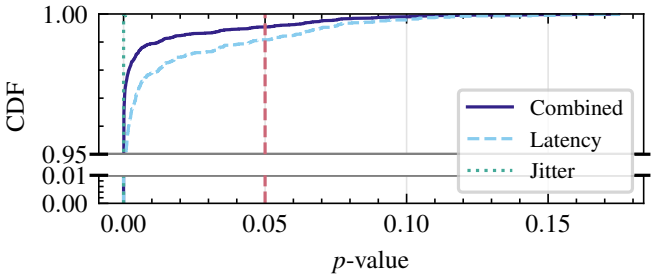


Fig. 3: The p-values of the ADF test for all flows and topologies. For latency, jitter, and both metrics combined.

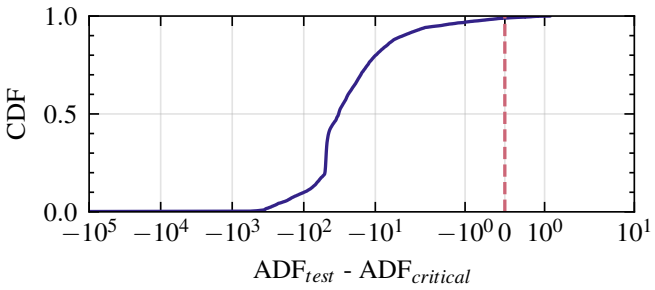


Fig. 4: The difference between the ADF test- and critical values. Negative values mean that the test value is smaller than the critical value, indicating stationary data.

B. Optimizations

Threshold selection is based on the stability of parameter estimations. Figure 5 shows the stability of the two parameters, the tail of the GPD ξ, and a modified version of the scale $\sigma^\star$. The scale is modified as shown in Equation (3) to de-couple it from the threshold [2].

$$\sigma^\star = \sigma_\mu - \xi \cdot \mu \tag{3}$$

Both ξ and $\sigma^\star$ behave roughly stable for a threshold of up to $\eta_{99.5}$.

C. Tail Model Verification

The GPD describing the tail behaves differently for different values of the tail parameter ξ. A value smaller than zero results in a bounded tail converging to a maximal value. A value larger than zero results in a non-converging unbounded tail. Table II shows the portion on bounded and unbounded tails for latency and jitter respectively. We can observe that a majority of latency tail models have an upper bound, whereas only a minority of jitter tail models do. Since jitter is a metric derived

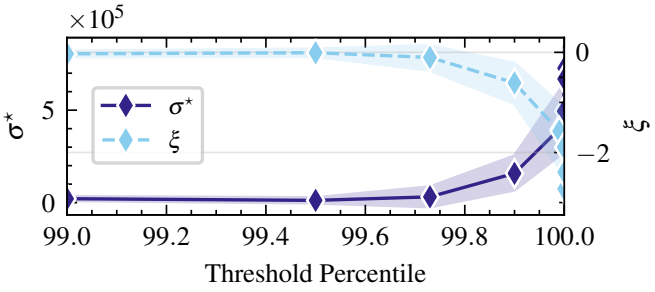


Fig. 5: Stability of MLE-derived GPD parameters over different percentile thresholds

TABLE II: Percentage of flows with bounded and unbounded tail behavior for latency and jitter values respectively

Metric	Bounded Tail	Unbounded Tail
Latency	3,507 (57.51%)	2,591 (42.49%)
Jitter	1,325 (21.73%)	4,773 (78.27%)

from latency, it should be as equally bounded as latencies are. This is not the case which indicates that the EVT approach is more suitable to model latencies. We will only consider the bounded tail latency models for the remainder of the evaluation.

The following shows results obtained by comparing latency predictions made by the flow level EVT models trained on 5% of the data and evaluated on 95% of the data.

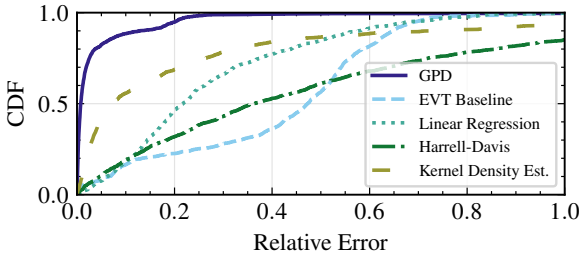
1) Percentiles: Table III shows the Median Absolute Percentage Error (MdAPE) between GPD and evaluation data percentiles. Figure 6 shows the relative error between GPD and evaluation data percentiles for four selected percentiles. Furthermore, it contains a comparison to four other methods. For the 50^{th}, 90^{th}, and 99.999^{th} percentiles the GPD outperforms all other methods. For the maximum, the GPD outperforms all methods except for the EVT baseline.

To get an understanding of whether GPD models are indeed flow specific, as assumed thus far, we compare flow-level percentiles to percentiles derived from GPD models on a network-level, i.e., aggregated over all flows traversing a network. Should the models not be flow specific, we would assume the network-level model to perform better since it has more data points available. Figure 7 shows the relative errors for different percentiles of flow- and network-level GPD models. The network-level models perform slightly better for low percentiles, whereas the flow-level models perform significantly better for high percentiles. We conclude that EVT models should be derived at the flow level when high percentiles of the tail are of importance.

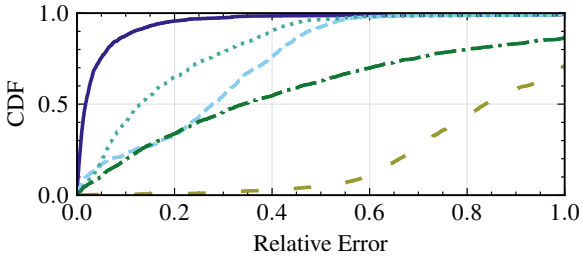
2) Return Levels and Exceedances: The return level x_m for the return period m is the value that is exceeded on average once during m. We compare the return levels for two

TABLE III: Median Absolute Percentage Error (MdAPE) of GPD predictions for different tail percentiles

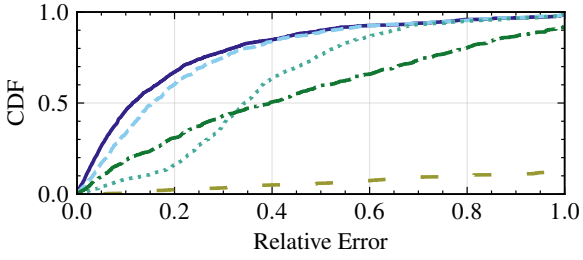
Percentile	50	75	90	99	99.9	99.99	99.999	100
MdAPE [%]	0.7	1.0	1.8	4.2	6.8	9.6	11.4	16.8



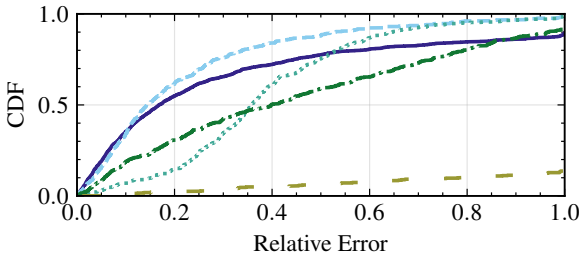
(a) 50^{th} percentile



(b) 90^{th} percentile



(c) 99.999^{th} percentile



(d) 100^{th} percentile

Fig. 6: Relative error in percentile predictions for different tail percentiles

periods: For 100% of the evaluation dataset (x_{100}) and 10% of the evaluation dataset (x_{10}). For the return level x_{100} we would expect one exceedance of this value on average in the complete evaluation dataset. For the return level x_{10} we would expect one exceedance in 10% of the evaluation dataset or ten exceedances in the complete evaluation dataset. Figure 8a shows the number of exceedances for the two return periods of flow-level models. We consider the number of exceedances to be the respectively expected one or ten if there is a return level within the confidence intervals that satisfies this constraint. For the 100% and 10% return levels, we obtain the correct number of exceedances for 75% and 85% of flows respectively. Comparing these results to network-level models is shown

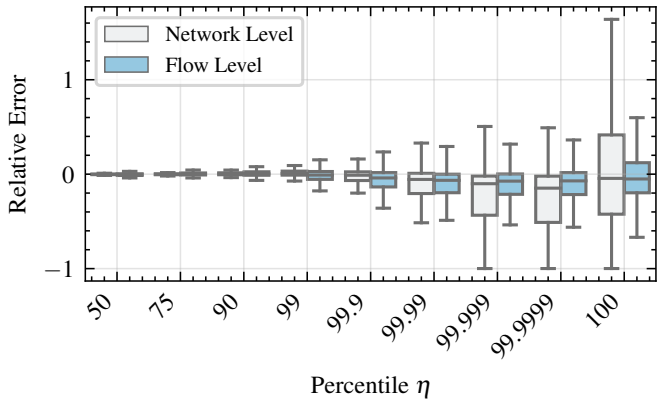


Fig. 7: Relative error of the GPD predictions on the flow- and network level for different percentiles

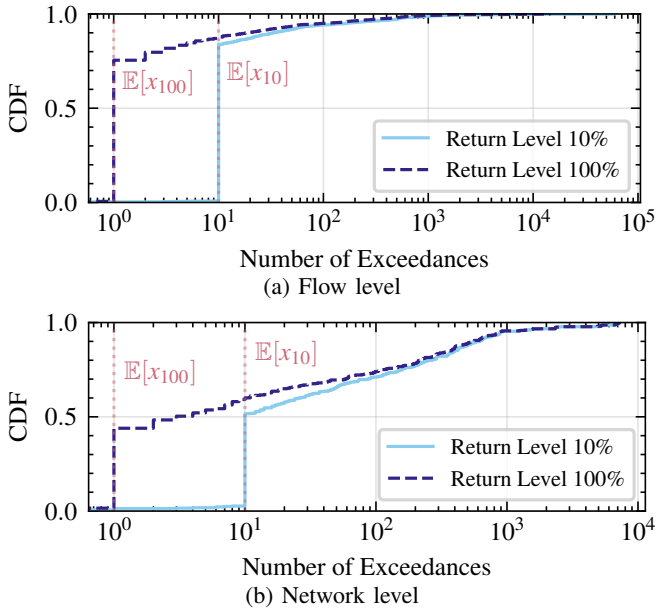


Fig. 8: Return levels and associated number of exceedances within confidence intervals

in Figure 8b. We can observe significantly less predictive power. This matches the observations for the relative error of percentile values, showing that flow-level models are better suited than network-level models, despite containing less data.

D. Limitations

The quality of EVT models and predictions depends on the amount of available data as well as on the confidence level of the distribution fitting and return level calculation. Furthermore, latencies of flows were mostly stationary in this setup but might not be stationary in general.

V. Conclusion

We showed that EVT can be applied to predict flow-level end-to-end latencies in different virtualized multi-hop networks connected over physical wires. The models were evaluated on larger time horizons compared to related work. The predictive power of tail percentiles was shown to exhibit a small relative error, and exceedances of latency values were predicted with an accuracy of 75-85% for a two- and twentyfold time horizon respectively. Future work includes scaling to larger topologies as well as including specific types of topologies such as leaf-spine topologies.

Acknowledgment

This work was supported in part by the European Union Horizon 2020 research and innovation programme (project SLICES-SC, 101008468), the Bavarian Ministry of Economic Affairs, Regional Development and Energy (project 6G Future Lab Bavaria), and the German Federal Ministry of Education and Research (project 6G-life, 16KISK001K, and project 6G-ANNA, 16KISK107).

References

[1] L. De Haan, A. Ferreira, and A. Ferreira, *Extreme Value Theory: An Introduction*. Springer, 2006, vol. 21.

[2] S. Coles *et al.*, *An Introduction to Statistical Modeling of Extreme Values*. Springer, 2001, vol. 208.

[3] I. Rached and E. Larsson, "Tail Distribution and Extreme Quantile Estimation using Non-Parametric Approaches," in *High-Performance Modelling and Simulation for Big Data Applications*. Springer, 2019, pp. 69–87.

[4] S. B. Provost *et al.*, "On the q-Generalized Extreme Value Distribution," *REVSTAT-Statistical Journal*, vol. 16, no. 1, pp. 45–70, 2018.

[5] "TS 22.104 5G Service Requirements for Cyber-Physical Control Applications in Vertical Domains," Sep. 2020. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/122100_122199/122104/16.05.00_60/ts_122104v160500p.pdf

[6] F. J. Cazorla, T. Vardanega, E. Quiñones, and J. Abella, "Upper-bounding Program Execution Time with Extreme Value Theory," in *13th International Workshop on Worst-Case Execution Time Analysis*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

[7] T. Hsing, "On Tail Index Estimation using Dependent Data," *The Annals of Statistics*, pp. 1547–1569, 1991.

[8] N. Mehrnia and S. Coleri, "Wireless Channel Modeling based on Extreme Value Theory for Ultra-reliable Communications," *IEEE Transactions on Wireless Communications*, 2021.

[9] M. Bennis *et al.*, "Ultrareliable and Low-Latency Wireless Communication: Tail, Risk, and Scale," *Proceedings of the IEEE*, vol. 106, no. 10, pp. 1834–1853, 2018.

[10] C.-F. Liu *et al.*, "Dynamic Task Offloading and Resource Allocation for Ultra-Reliable Low-Latency Edge Computing," *IEEE Transactions on Communications*, vol. 67, no. 6, pp. 4132–4150, 2019.

[11] Y. Zhu *et al.*, "Reliability-Optimal Offloading in Low-Latency Edge Computing Networks: Analytical and Reinforcement Learning Based Designs," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 6058–6072, 2021.

[12] F. Wiedner *et al.*, "HVNet: Hardware-Assisted Virtual Networking on a Single Physical Host," *IEEE INFOCOM WKSHPS CNERT 2022*, 2022.

[13] ——, "HVNet: Hardware-Assisted Virtual Networking on a Single Physical Host," 2022. [Online]. Available: https://mediatum.ub.tum.de/1638129

[14] L. Santinelli, J. Morio, G. Dufour, and D. Jacquemart, "On the Sustainability of the Extreme Value Theory for WCET Estimation," in *14th International Workshop on Worst-Case Execution Time Analysis*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2014.

[15] D. Griffin and A. Burns, "Realism in Statistical Analysis of Worst Case Execution Times," in *10th International Workshop on Worst-Case Execution Time Analysis (WCET 2010)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2010.

[16] I. P. Lemos *et al.*, "thresholdmodeling package," Feb. 2020. [Online]. Available: https://github.com/iagolemos1/thresholdmodeling

[17] P. Friederichs and T. L. Thorarinsdottir, "Forecast Verification for Extreme Value Distributions with an Application to Probabilistic Peak Wind Prediction," *Environmetrics*, vol. 23, no. 7, pp. 579–594, oct 2012. [Online]. Available: https://doi.org/10.1002%2Fenv.2176