Unevenly Spaced Time Series from Network Traffic

1st Josef Koumar
Czech Technical University in Prague
Prague, Czech republic
koumajos@fit.cvut.cz, 0000-0002-3643-9723

2nd Tomas Čejka
CESNET, a.l.e.
Prague, Czech republic
cejkat@cesnet.cz, 0000-0001-7794-9511

***Abstract*—Reliable detection of security events is essential for network security. Therefore, a suitable traffic representation and model are required. Contrary to the currently used approaches, this paper presents Unevenly Spaced Time Series (USTS) as a feasible representation of network traffic with several brilliant benefits for analysis. The article concerns several types of USTS. A dataset captured on a real ISP network was created to evaluate the properties of USTS. The dataset contains over 35 million time series. We experimentaly proved the USTS is suitable for network traffic analysis and allow automatic processing, e.g., to classify network traffic.**

***Index Terms*—time series, network traffic, packets, IP flows**

I. Introduction

Time series (TS) are essential sources of information for traffic analysis. Time-related features and behavior of the traffic can help to deal with the decreased visibility into the network traffic since they can be studied regardless of the encryption. Researchers and engineers currently use Evenly spaced time series (ESTS) created by aggregating network traffic in time intervals. For example, Cook et al. [1] address anomaly detection methods for IoT TS. They mention many articles that use ESTS and describe challenges with aggregation, noise and non-stationarity of TS, i.e. removing seasonality and trend. However, ESTS can cause an information loss that can affect the performance of the analysis.

Therefore, this paper targets the topic using the so-called Unevenly spaced time series (USTS). This type of TS is a natural representation of network traffic because one datapoint is one IP flow or packet with time defined by transmission. Also, they do not cause loss of information because they do not aggregate network traffic. The contributions of our work can be summarized as follows:

- We show several types of USTS from network traffic, their comparison with ESTS, and describes their benefits.
- We experimentally proved the stationarity of USTS from network traffic, which allows their automatic processing.
- We experimentally proved that USTS from network traffic occurs with periodic behaviors implying predictability.
- We create a dataset of 35 million USTS from a real high-speed network CESNET2[1].

This research was funded by the Ministry of Interior of the Czech Republic, grant No. VJ02010024: Flow-Based Encrypted Traffic Analysis and also by the Grant Agency of the CTU in Prague, grant No. SGS23/207/OHK3/3T/18 funded by the MEYS of the Czech Republic.

[1]the Czech national research and education network

978-3-903176-58-4 ©2023 IFIP

II. Related Works

The USTS naturally occurs in many industrial and scientific domains like astronomy, medicine, economics, etc. There are a lot of methods and tools for analyzing USTS, for example, detection of periodicity [2]–[4], spectral analysis methods [5], or anomaly detection [6]. There are also existing methods for training the neural networks based on USTS [7], [8].

The TS from network traffic are used, e.g., for network traffic monitoring, anomaly detection and traffic prediction. The most commonly used TS are created by aggregating the traffic of some process (mostly one network device) during a specific time window into a single value. As a result, an ESTS is created. However, the application of USTS is almost missing in the network traffic analysis domain.

In the article [9], the authors use ESTS to examine the inter-packet gaps of adjacent packets. Network traffic is also sampled into multiple sampling intervals to get ESTS for chaotic characteristic analysis in paper [10]. Also, the papers [11]–[14] use ESTS for network traffic prediction. Moreover, articles [9], [15]–[17] use ESTS for anomaly detection. For classification by deep learning, the articles [18]–[21] use the ESTS, where time differences are used.

The existing related works mostly use TS from network traffic based on *regularly aggregated values* over specific time intervals or USTS that are aggregated into ESTS. Contrary, this paper describes USTS from network traffic created without any aggregation or interpolation. We also show that USTS can be exploited for classification, anomaly detection, or traffic prediction.

III. Time Series Analysis (TSA)

Most often, TS are considered with evenly spaced time between observations. This type of TS is called *Evenly spaced time series (ESTS)*, also called regularly sampled or uniformly sampled. They are defined as the sequence of observation $\{X_n\} = \{x_1, \ldots, x_n\}$ taken in times $\{T_n\} = \{t_1, \ldots, t_n\}$ which satisfy the equation $t_{j+1} - t_j = t_j - t_{j-1}, \forall j \in 2, \ldots, n-1$. There are also types of TS where the times of observations are paced by a monitored process and cannot be chosen. However, these TS do not have the same times which satisfy $t_{j+1} - t_j = t_j - t_{j-1}$. That means the times are, in general, not regularly spaced, that means, $\delta_j = t_{j+1} - t_j, \forall j \in \{1, \ldots, n-1\}$, is not constant. They are called *Unevenly spaced time series (USTS)*, also called unequally spaced, or irregularly sampled.

The basic model used for modelling ESTS is an additive model. The additive model of ESTS is defined as follows:

$$x_t = \mathcal{T}_t + \mathcal{S}_t + \epsilon_t \tag{1}$$

The $x_t \in \{X_n\}$ is an observation taken in time $t \in \{T_n\}$, $\mathcal{T}$ is the trend component, $\mathcal{S}$ is the seasonal component, and ϵ is an irregular component, that represents the noise or random component. If we want to perform a TSA, then we must extract trend and seasonal components beforehand. This process is called Time Series Decomposition (TSD). However, without TSD, we cannot get a suitable model of the TS. Furthermore, TSD is usually done by hand, and it is problematic to process it automatically.

IV. Creating a USTS from Network Traffic

We want to directly analyze one specific process or application running on a particular device. At first, we divide network traffic into so-called network dependencies [22] that are long-term communication of pairs of devices, where one device provides service to another. Network dependencies merge multiple packets or flows into a single TS. Each packet or flow contains a timestamp of the transmission time, which precisely determines its position in the TS. Creating such a sequence results in a USTS.

Suppose we aggregate network traffic in time intervals as it is used in most published papers. In such a case, we get TS with high dependence on the aggregation interval. For example, the ESTS was created with aggregation interval 60 seconds in Fig. 1. However, we can see that the explicit cyclic behavior of flows in USTS is noised by aggregation. Furthermore, the aggregation of USTS often causes zero values in ESTS, whereas zero values are usually a problem for TSA because known mathematic methods and tools for ESTS cannot work with them very well.

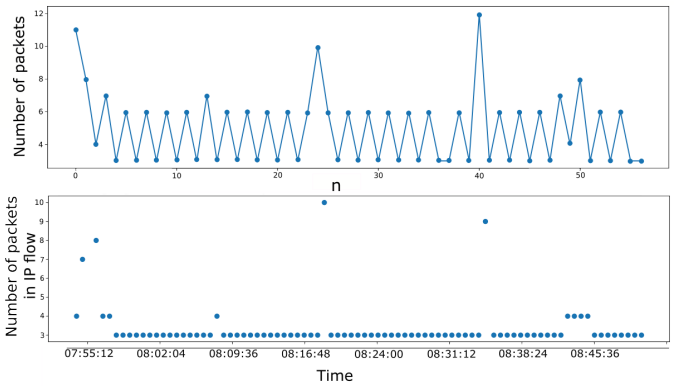

Fig. 1. Comparison of ESTS (top graph) and USTS (bottom graph)

We define multiple types of USTS from network traffic, designated by what represents one data point and what data points have in common.

A. Packet Time Series

A Packet Time Series (PTS) from network traffic is TS, where a data point represents a network packet. Furthermore, PTS is a univariate TS with a variable number of bytes in the network packet. The time information $t_i \in \mathbb{R}$ of i-th data point is defined by the transmission time of i-th packet. So PTS are inevitably USTS.

B. Flow Time Series

A flow represents aggregated information from a sequence of packets with the same attributes in the packet headers. The Flow Time Series (FTS) are multi-dimensional to cover all valuable information, such as the number of packets and bytes. Therefore, such FTS are multivariate USTS. Another important fact is that the flow record contains two timestamps, namely the time of the first and the last packet of the flow, so the FTS has two time axes. These two time axes can generate additional FTS variables, which can be well applied in analysis and possible classification. The first axis is a variable duration of the flow. The second one is a variable time difference representing the gaps between flows.

C. Single Flow Time Series (SFTS)

Since PTS may contain packets of any connection together, creating separate TS is useful. Such TS that contains a packet sequence of just a single flow is called a Single Flow Time Series (SFTS). Because a SFTS is a special case of PTS of one flow only, it cannot contain network traffic of more than one process, and the only noise that can occur is packet retransmission. Therefore, the noise in the SFTS is minimal.

D. Model of USTS from network traffic

We can model all types of USTS with a single equation:

$$x_t = \mathcal{S}_t + \epsilon_t \tag{2}$$

In comparison with the additive model of ESTS shown in the equation (1), the trend component $\mathcal{T}$ is absent in the equation (2). If the trend component was present in the USTS, then there is a permanent increase in the number of bytes. Thus, all fluctuations seeming like a trend component in USTS are noise. Nevertheless, PTS can have some local trends as characteristics. We must also consider the time axis of data points. Experiments proved that all types of USTS have times that act like *Random walk*. So, we can use *Random walk model* with the model shown in Equation (2) and model USTS for anomaly detection or TS forecasting.

V. Characteristics of Unevenly Spaced Time Series from Network Traffic

We have created three datasets for experiments with the USTS from network traffic. The first dataset contains 2,6 million FTS created from 259 million flows, 19 million PTS created from 110 million packets, the second dataset contains, and the third dataset contains 15 million SFTS created from 160 million packets. These datasets were created by traffic capture on the ISP infrastructure of the CESNET2. The datasets [23] and jupyter notebooks[2] with more experiments are published with this paper.

[2]https://github.com/koumajos/USTS

A. Stationarity

According to [24], properties of a stationary TS do not depend on the time of observation. The TS with a trend or with seasonality is not stationary, but the TS with periodic (or cyclic) behavior can be stationary. From that, we can hypothesize that PTS, FTS, and SFTS are stationary because they do not have a trend, and seasonal component S can be a periodic (or cyclic) behavior. We tested the stationarity using the Augmented Dickey-Fuller (ADF) test as described in the paper [25]. Results of our experiments are shown in Table I.

TABLE I
STATIONARY TEST APPLIED ON PTS, FTS, AND SFTS DATASETS (IN %)

	Time	Number of data points					
	[min]	all	<25	25-100	100-500	500-1000	≥1000
FTS	**all**	85.83	72.26	91.30	97.37	99.64	99.66
	< 1	67.08	66.39	86.62	85.71	100.0	NaN
	1-10	56.73	55.66	76.88	93.94	91.67	88.89
	10-60	65.66	62.29	83.24	91.67	97.14	100.0
	≥ 60	88.13	75.32	91.58	97.39	99.64	99.66
PTS	**all**	83.85	82.57	85.39	90.72	95.38	95.32
	< 1	76.47	74.91	83.52	89.67	92.65	89.07
	1-10	85.67	86.23	83.38	87.85	93.15	94.95
	10-60	95.86	98.27	91.48	93.19	97.77	98.54
	≥ 60	85.85	81.63	82.58	87.40	94.58	96.70
SFTS	**all**	54.97	50.36	83.91	89.01	94.57	97.22
	< 1	45.24	41.06	81.77	87.13	94.72	96.94
	1-2	72.48	71.11	70.50	81.10	92.53	97.59
	2-4	69.16	66.99	70.74	83.77	90.53	97.78
	≥ 4	82.05	78.26	91.45	94.02	95.64	97.58

According to the results, FTS, PTS, and SFTS become stationary in a relatively short period. Furthermore, the results are strongly affected by the number of data points in the USTS and the duration of the USTS. Almost every PTS and FTS with at least 500 data points and with a duration of at least 1 hour is stationary. Thus, our assumption of the USTS model in Equation (2) that model has no trend component is correct, and the seasonal component represents a periodic (or cyclic) behavior of the USTS. Moreover, we do not need to do the TSD of USTS from network traffic, i.e., removing trend and seasonality, before performing analysis. This is crucial because such TS preprocessing must be usually done before proper use. Furthermore, it must be processed by a human in most situations. So stationarity enables us to use mathematical methods and tools for USTS and automatically process network traffic for security threat detection.

B. Hurst exponent

As part of our analysis and experiments focused on the behavior of USTS from network traffic, we performed tests using the Hurst exponent [26] calculated by the procedure for USTS by *Ji et al* [27]. If the Hurst exponent $H \in \langle 0; 0.5)$, then it indicates a long-term switching between high and low values in adjacent pairs and the TS is anti-persistent. If $H \sim 0.5$, then this indicates a random (uncorrelated) TS. Furthermore, if $H \in (0.5; 1\rangle$, then it indicates a long-term positive autocorrelation in the TS and the TS is persistent.

The histogram with results of the experiments is shown in Fig. 2, where the red marked interval is for random TS. We filter TS, which have at least 30 s time length and 20 datapoints, and then FTS, PTS, and SFTS tend to have a long-term positive autocorrelation.

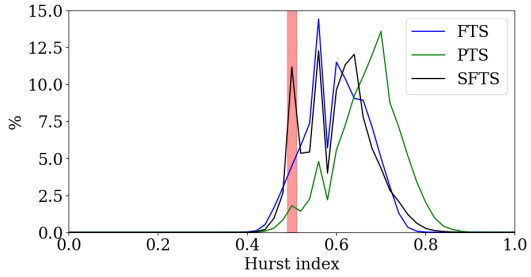

Fig. 2. Hurst exponent of TS with time length at least 30 s and 20 datapoints

Therefore from experiments with the stationary TS, we can confirm that the model of USTS from network traffic in Equation (2) is suitable. Moreover, from the Hurst exponent, we also know that they tend to be periodic, so the seasonal component $\mathcal{S}$ represents the periodic (or cyclic) behavior. So, this leads to USTS from network traffic being well predictable and suitable for anomaly detection and network traffic forecasting.

VI. Application of USTS on Real Traffic

The presented USTS approach was used for network traffic analysis in our previous work focused on network traffic classification in practice. The classifier was based on periodic behavior detection using Lomb-Scargle periodogram applied on FTS to detect periodicity. The experiments and results of classification of 61 applications, services and operation systems with 90% F1-score were described in detail in [22].

VII. Conclusion

In this paper, we have presented i) several types of USTS from network traffic (PTS, FTS, and SFTS), ii) their characteristics that were experimentally proved, iii) model in Equation (2) and time axis behavior. The results of our experiments show that USTS are feasible for network traffic analysis and exhibits significant advantages over ESTS, which are as follows: 1) TS distribution are not affected by aggregation interval, 2) we know what data points and their values represent, 3) it is not necessary to set aggregation time interval, which is hard to select, 4) there are no zero values (times without data points), 5) they contain minimal noise, 6) they are stationary, so there is no need to perform TSD before TSA, which allows automatic procession, and 7) they usually occur with clear periodic behavior.

To conclude this article, the USTS from network traffic and their model can benefit many applications, such as anomaly detection, traffic forecasting, periodicity detection, network traffic classification, and detection of security threats based on their behaviors. However, the computation efficiencies of USTS can be problematic in high-speed networks. In our future work, we will compare USTS and ESTS methods in precision and deployment in real-time networks.

REFERENCES

[1] Andrew A. Cook, Göksel Mısırlı, and Zhong Fan. Anomaly detection for iot time-series data: A survey. *IEEE Internet of Things Journal*, 7(7):6481–6494, Jul 2020.

[2] James H Horne and Sallie L Baliunas. A prescription for period analysis of unevenly sampled time series. *The Astrophysical Journal*, 1986.

[3] N.R. Lomb. Least-squares frequency analysis of unequally spaced data. 1976.

[4] Jeffrey Scargle. Studies in astronomical time series analysis. ii - statistical aspects of spectral analysis of unevenly spaced data. *The Astrophysical Journal*, 1982.

[5] Adolf Mathias et al. Algorithms for spectral analysis of irregularly sampled time series. *Journal of Statistical Software*, 2004.

[6] Yang Jiao et al. Timeautoml: autonomous representation learning for multivariate irregularly sampled time series. *arXiv preprint arXiv:2010.01596*, 2020.

[7] Brett Naul et al. A recurrent neural network for classification of unevenly sampled variable stars. *Nature Astronomy*, 2(2):151–155, 2018.

[8] Chenxi Sun et al. A review of deep learning methods for irregularly sampled medical time series data. *preprint arXiv:2010.12493*, 2020.

[9] Jarosław Bernacki et al. Anomaly detection in network traffic using selected methods of time series analysis. *IJCNIS*, 2015.

[10] Zhongda Tian. Chaotic characteristic analysis of network traffic time series at different time scales. *Chaos, Solitons & Fractals*, 130, 2020.

[11] Hao Yin et al. Network traffic prediction based on a new time series model. *International Journal of Communication Systems*, 2005.

[12] Sangjoon Jung et al. A prediction method of network traffic using time series models. In *ICCSA*, 2006.

[13] Rishabh Madan et al. Predicting computer network traffic: a time series forecasting approach using DWT, ARIMA and RNN. In *IC3*, 2018.

[14] Theyazn HH Aldhyani et al. Intelligent hybrid model to enhance time series models for predicting network traffic. *IEEE Access*, 8, 2020.

[15] Huang Kai et al. Network anomaly detection based on statistical approach and time series analysis. In *AINA*. IEEE, 2009.

[16] Peter Kromkowski et al. Evaluating statistical models for network traffic anomaly detection. In *SIEDS*. IEEE, 2019.

[17] Łukasz Saganowski et al. Time series forecasting with model selection applied to anomaly detection in network traffic. *Logic Journal of the IGPL*, 2020.

[18] Ly Vu et al. Time series analysis for encrypted traffic classification: A deep learning approach. In *ISCIT*. IEEE, 2018.

[19] Amir M. Sadeghzadeh et al. Adversarial network traffic: Towards evaluating the robustness of deep-learning-based network traffic classification. *IEEE TNSM*, 2021.

[20] Mohammadreza MontazeriShatoori et al. Detection of DoH Tunnels using Time-series Classification of Encrypted Traffic. In *2020 IEEE Intl Conf: DASC/PiCom/CBDCom/CyberSciTech*, pages 63–70, 2020.

[21] Jan Luxemburk and Tomáš Čejka. Fine-grained tls services classification with reject option. *arXiv preprint arXiv:2202.11984*, 2022.

[22] Josef Koumar and Tomáš Čejka. Network traffic classification based on periodic behavior detection. *CNSM*, 2022.

[23] Josef Koumar and Tomáš Čejka. CESNET-USTS23: a benchmark dataset of Unevenly spaced time series from network traffic, May 2023. Zenodo: https://doi.org/10.5281/zenodo.7923744.

[24] Denis Kwiatkowski et al. Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? *Journal of Econometrics*, 1992.

[25] Rizwan Mushtaq. Augmented dickey fuller test. 2011.

[26] A. A. Anis and E. H. Lloyd. The expected value of the adjusted rescaled hurst range of independent normal summands. *Biometrika*, 63(1):111–116, 1976.

[27] Li-Jun Ji, Wei-Xing Zhou, Hai-Feng Liu, Xin Gong, Fu-Chen Wang, and Zun-Hong Yu. R/s method for unevenly sampled time series: Application to detecting long-term temporal dependence of droplets transiting through a fixed spatial point in gas–liquid two-phase turbulent jets. *Physica A: Statistical Mechanics and its Applications*, 388(17):3345–3354, 2009.