Towards a Cost vs. Quality Sweet Spot for Monitoring Networks

Nofel Yaseen[‡◇], Behnaz Arzani[‡], Krishna Chintalapudi[‡], Vaishnavi Ranganathan[‡],
Felipe Frujeri[‡], Kevin Hsieh[‡], Daniel S. Berger[‡], Vincent Liu[◇], Srikanth Kandula[‡]
Microsoft[‡] and University of Pennsylvania[◇]

ABSTRACT

Continuously monitoring a wide variety of performance and fault metrics has become a crucial part of operating large-scale datacenter networks. In this work, we ask whether we can reduce the costs to monitor – in terms of collection, storage and analysis – by judiciously controlling how much and which measurements we collect. By positing that we can treat almost all measured signals as sampled time-series, we show that we can use signal processing techniques such as the Nyquist-Shannon theorem to avoid wasteful data collection. We show that large savings appear possible by analyzing tens of popular measurement systems from a production datacenter network. We also discuss some challenges that must be solved when applying these techniques in practice.

ACM Reference Format:
Nofel Yaseen[‡◇], Behnaz Arzani[‡], Krishna Chintalapudi[‡], Vaishnavi Ranganathan[‡], Felipe Frujeri[‡], Kevin Hsieh[‡], Daniel S. Berger[‡], Vincent Liu[◇], Srikanth Kandula[‡] . 2021. Towards a Cost vs. Quality Sweet Spot for Monitoring Networks. In *The Twentieth ACM Workshop on Hot Topics in Networks (HotNets '21), November 10–12, 2021, Virtual Event, United Kingdom.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3484266.3487390

1 INTRODUCTION

High availability guarantees are etched into the service-level agreements of data centers, and failure to meet them has significant monetary impact. As a result, data centers deploy large-scale monitoring systems that continuously monitor various performance metrics to help quickly identify (or predict) and alleviate service disruptions. For the most common class of monitoring systems, typical operation entails periodically sampling operational parameters of various data center components (e.g., CPU temperatures, packet drops, path latencies); operators collect and analyze these measurements in real-time or store them for later analysis.

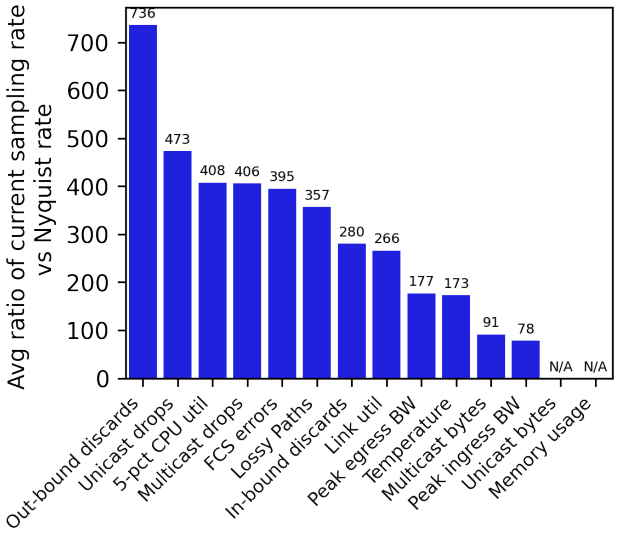


Figure 1: How far above the Nyquist rate current monitoring systems operate on average.

At the scale of modern data centers, monitoring systems incur significant costs in terms of storage, network bandwidth, and CPU resources [2, 9, 16, 25, 26, 32, 35]. For a large class of measurements, e.g. interface counters, ping latencies, traceroutes, or results from sketches, there is often a quality versus cost tradeoff. Obtaining measurements more frequently provides potentially "higher quality" monitoring but also places greater demands on data center resources, i.e. has higher costs.

We ask: "what is the right frequency at which a measurement must be taken?" We find that measurements are being conducted at a high frequency, conservatively, and without the benefit of a systematic analysis. In our survey of measurements collected from many production monitoring systems of a large cloud provider, we observe that admins typically choose to err on the safer side; that is, they collect as much information as possible subject to some (arbitrarily set) resource constraints. They often cannot answer whether the chosen measurement rate is adequate for the given metric or whether measuring more (or less) frequently will lead to better (or no worse) insights? Also, admins often express concern that collecting less information could lead to missing out on important insights.

HotNets '21, November 10–12, 2021, Virtual Event, United Kingdom

ACM ISBN 978-1-4503-9087-3/21/11…$15.00
https://doi.org/10.1145/3484266.3487390

Information theory provides techniques that can help with this question. For instance, the Nyquist sampling frequency [18] (Nyquist rate) of a bandlimited[1] signal determines the minimum sampling frequency required to capture the signal without any information loss. Intuitively, the Nyquist rate is a measure of "how quickly" a signal changes in time: measurements need not be taken faster than the underlying value changes.

Our key contribution is a somewhat surprising result obtained after estimating the Nyquist sampling frequencies of a range of measurements in production data centers— in many cases, the conservatively chosen sampling rates currently employed in production are orders of magnitude greater than their actual Nyquist rates. Figure 1 is a summary of our observations showing that operators are significantly over-sampling a diverse set of metrics (methodology in Section 3.2). This indicates that many measurements could be sampled far less frequently, resulting in significant reduction in monitoring costs with "negligible" loss in information (see Section 4). In Section 3.2 we quantify this potential cost reduction on many real-world metrics.

In practice, the Nyquist rate of a measurement may vary over time or be different at different devices. During relatively quiescent periods, a measurement may not change "quickly" and a low sampling rate would suffice. However, at other periods, the signal may be more dynamic and require sampling at a higher rate. A practical system must therefore continuously analyze the measurements to determine whether the currently employed sampling rate is sufficient. It must also be able to track changes in the Nyquist rate and adapt the measurement rate accordingly. To this end we propose a dynamic sampling method in Section 4. We show preliminary results that demonstrate the effectiveness of this approach in our extended version [31].

While signal processing techniques such as compressive sensing and sparse FFT have been applied before, to our knowledge, we are unaware of any prior work that applies the Nyquist principle to find appropriate measurement rates for datacenter metrics. Our initial results are promising and point to large untapped gains.

2 A PRIMER ON NYQUIST–SHANNON

Signals are functions of one or more independent variables, with the primary independent variable in most signals being time. We can further divide these signals into two categories: continuous- and discrete-time signals where the difference is in the domain of the function. More concretely, continuous-time signals are functions, $f(t)$, where $t : \mathbb{R} \rightarrow \mathbb{R}$, and discrete-time signals are functions, $f(T)$, where $T : \mathbb{N} \rightarrow \mathbb{R}$.

[1]A bandlimited signal is such that the signal and all its derivatives are continuous in time.

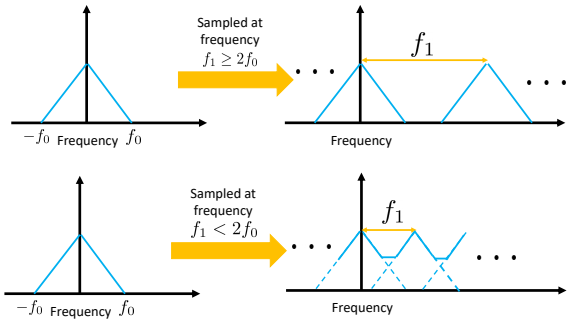


Figure 2: Showing the result, in the frequency domain, when sampling above and below the Nyquist rate. Sampling a signal at frequency f_1 and reconstructing it can be thought off, in the frequency domain, as adding copies of the signal which are f_1 apart.

The outputs of today's measurement systems are discrete-time signals. Sampling a signal converts continuous-time signals into discrete-time signals or down-samples discrete-time signals to reduce the costs to monitor and store telemetry.

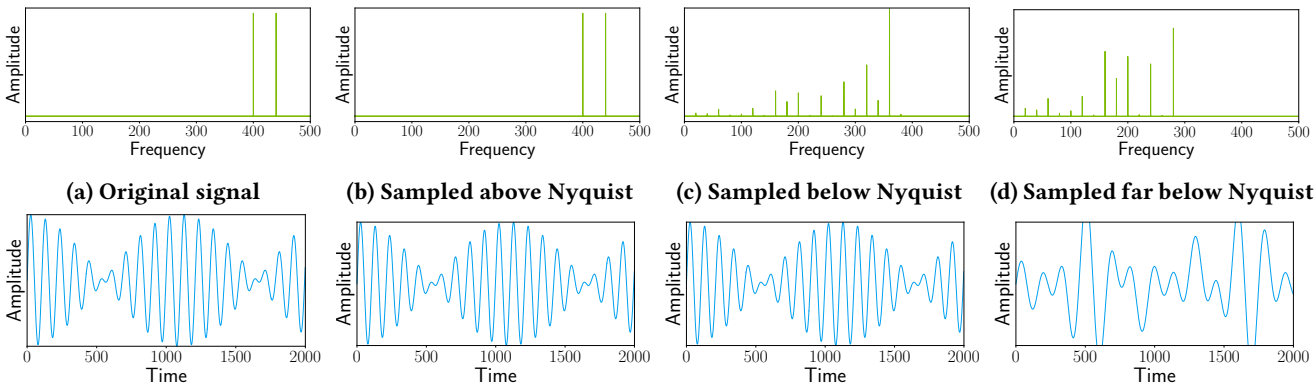
The Fourier transform. We can convert signals from functions over time into functions over frequency. These conversions from the time to frequency domain are *lossless*. The procedure to accomplish this translation is the Fourier transform, which produces a function whose magnitude at frequency f is the amount of that frequency present in the original function. An Inverse Fourier Transform converts the signal back to the time-domain.

The Fast Fourier Transform (FFT) is an efficient algorithm that applies over discrete sampled signals. Given an input N, the FFT divides the frequency space from 0 to the maximum frequency in the signal into N discrete bins and computes the signal power in each bin. The square of these per-bin magnitudes is the Power Spectral Density (PSD).

Nyquist rates and Fourier transforms in practice. The *Nyquist–Shannon theorem* states: if a function $x(t)$ contains no frequencies higher than f_0, then, sampling it at a rate at or above $2f_0$ ensures the original signal can be recovered completely; we call $2f_0$ the *Nyquist rate* of $x(t)$ [18].

When the sampling frequency is below the nyquist rate, *aliasing* occurs (bottom half of Figure 2) which distorts the PSD and prevents recovering the original signal. Figure 3 demonstrates these effects on an example signal.

The implication of this theorem in practice is that when a monitored signal is sampled at or above the Nyquist rate of that signal, then operators can rest assured no information is being lost due to sampling. On the other hand, when aliasing occurs, the extent of the information loss depends on the difference between the PSD of the aliased signal and that of the original. The impact of information loss depends on



(a) Original signal (b) Sampled above Nyquist (c) Sampled below Nyquist (d) Sampled far below Nyquist

(e) Original signal (f) Reconstructed: Figure 3b (g) Reconstructed: Figure 3c (h) Reconstructed: Figure 3d

Figure 3: A signal and its sampled versions in the frequency domain (top row). The original signal (bottom) is the superposition of two sin waves at 400 and 440 Hz. The time-domain representations of the sampled versions are reconstructed and upsampled. Variants shown are: (a) the original signal, (b) the signal sampled at above the Nyquist rate (890 Hz), (c) the signal sampled at slightly below the Nyquist rate (800 Hz), and (d) the signal sampled at significantly below the Nyquist rate (600 Hz). Aliasing is observable in the frequency domain of (c) and (d).

the application using the measurements and operators must determine what level of aliasing (if any) is acceptable.

3 QUANTIFYING THE OPPORTUNITY

Through this new lens of viewing existing network monitoring systems as collecting (sensing) signals, we can optimize existing monitoring systems. While we may need to address many challenges before realizing these techniques in practice (see Section 4), as a first step, we answer the question: what do we stand to gain if we use them?

We examine the network of a major cloud provider and the monitoring systems it has deployed. We describe why many of the existing monitoring systems this provider (and others similar to it) have deployed are sub-optimal. Here, we restrict our focus to monitoring systems that periodically poll a numeric metric. Later, we introduce opportunities for taking these observations further and applying them to other types of monitoring systems our community has developed.

3.1 Today's Monitoring Systems

Every aspect of the task of monitoring—collection, transmission, analysis, and storage—all consume resources that, when considering the scale of modern data centers, represent a non-negligible overhead. In an effort to reduce the costs of monitoring, for non-event-based data, today's systems generally sample their target metrics periodically/randomly so as to capture the gist of the metric without sacrificing too many resources. Examples include systems that periodically poll switch counters [10, 32, 35], sample packets to construct flow records [5, 21], or send packets through the network to extract the instantaneous latency of the network [9, 26].

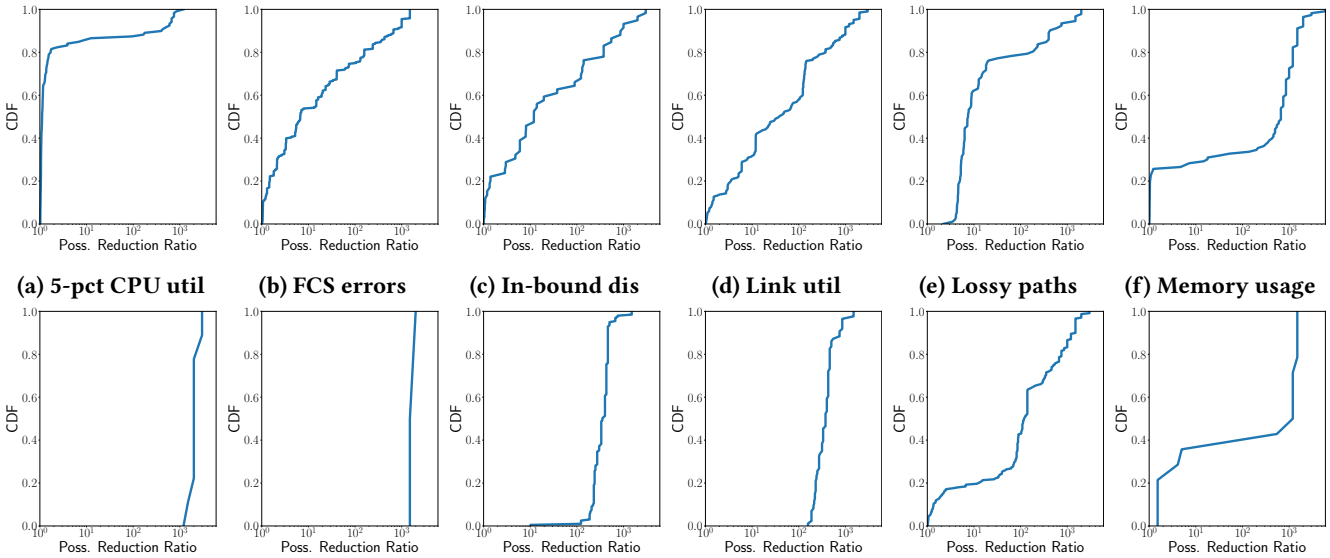
The sampling rate for the majority of these systems is entirely arbitrary [2, 35]. Often, the sampling rates are not governed by signal processing principles but chosen based on defaults and vague 'gut feelings' about the desired granularity of the data and the system and network-level overheads. For such monitoring systems aliasing effects are never evaluated and the chosen granularities are never re-considered.

The end result is that most of these systems are either (a) over-sampling (increasing overheads and wasting resources) or (b) under-sampling without any idea of how much information is lost. This has significant implications on the resource usage of existing monitoring systems.

3.2 Case Study: A Large Cloud Provider

We demonstrate empirically the opportunity afforded by applying the Nyquist-Shannon theorem. The monitoring systems we study encompass a wide range of monitoring systems in a large cloud provider including device temperature, packet drops, FCS errors [37], and link utilization.

Our goal is to identify the Nyquist rate of each of the signals these systems monitor. To do so: (a) for a given trace (where each trace is the data for a metric being measured on a single device), we compute the FFT and compute the total energy in the signal—the sum of the PSD across all FFT bins; (b) we add the PSD components in each FFT bin until we reach 99% of the total energy in the signal computed in (a); if we need *all* bins of the FFT to achieve 99% of the total energy we assume the signal is probably already aliased;



(a) 5-pct CPU util (b) FCS errors (c) In-bound dis (d) Link util (e) Lossy paths (f) Memory usage

(g) Multicast bytes (h) Multicast drops (i) Peak egress BW (j) Peak ingress BW (k) Temperature (l) Unicast bytes

Figure 4: CDFs of the ratio between the actual sampling rate and the computed Nyquist rate. Note x axes is in log scale and $x = 10$ indicates $10\times$ over-sampling. Each datapoint is one day's worth of data from a distinct device. We do not show the cases where we cannot reliably detect the Nyquist rate.

(c) otherwise, we report twice the frequency at which we capture 99% of the total energy as the Nyquist rate.

The approach above can uncover the Nyquist rate of the underlying signal *if* the measured trace is sampled above its Nyquist rate. We find this to be the common case in our traces. The converse case is challenging because when aliasing has already happened in a measured signal, the method above does not reliably produce the Nyquist rate. We discuss how to detect aliasing (under-sampling) in Section 4. Our choice of the 99% cut-off on total energy is a workaround to compensate for measurement noise. Using a higher value such as 99.99% would increase our estimate of the Nyquist rate and reduce performance gains but, in our experience, does not necessarily lead to a lower reconstruction error since the delta that is being captured is often just the noise.

In practice, monitoring systems do not produce perfectly sampled signals—samples are not always spaced at equidistant points in time. In such situations, we pre-clean the signal using nearest neighbor re-sampling [27, 28]; that is, we add values for missing samples based on nearby samples.

Figure 4 reports for each measured statistic the CDF of the ratio between the current sampling and the Nyquist rate we identified through the above approach. The ratio indicates the degree to which we are currently oversampling the underlying signal. We observe that in 20% of the examples the sampling rate can be reduced by a factor of $\sim 1000\times$.

We studied 1613 metric and device pairs (14 distinct metrics). Of these, 89% were sampling at higher than their Nyquist

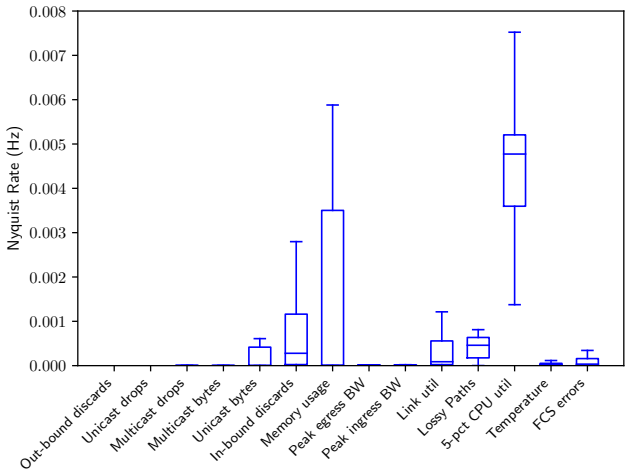


Figure 5: A box plot of the Nyquist rate of each system.

rate. Within a metric, the Nyquist rate varies across devices (Figure 5). For example, for the temperature signal, the Nyquist rate ranges from 7.99×10^{-7} Hz to 0.003 Hz. We also notice different Nyquist rates at different time periods on the same device. This indicates the properties of the underlying metric vary over time and across devices and the need for dynamically adapting the measurement rate. Adaptation must be quick because under-sampling would lead to aliasing and information loss. We discuss some relevant prior work and propose a dynamic sampling method in Section 4; we show an example of how the approach works in [31].

In our experiments the existing sampling rate is below the Nyquist rate of the underlying signal in about 11% of the metric-device pairs which require more careful inspection: it is possible existing monitoring systems must increase their sampling rates to fully capture these metrics. We also want to rule out issues such as measurement noise, data loss or data corruption that may have lead to an incorrect assessment of the Nyquist rate. We defer this to future work.

4 TOWARD DYNAMIC MONITORING

There is a significant opportunity to save resources in deployed monitoring solutions by analyzing the signals they produce. Acquiring the measurements may be a large expense and we discuss a new dynamic sampling method below to address this problem. The dynamic sampling method also applies when the Nyquist rate of a metric varies across devices or across time; we do not yet understand the reasons for these changes but nevertheless use dynamic sampling to ensure robustness to changes to the Nyquist rate of the underlying signal. We need to address a number of challenges in order to implement this method in practice. We next outline these challenges and potential solutions.

4.1 Detecting aliasing

A critical component to a solution is to detect when a chosen sampling rate has dipped below the Nyquist rate of the underlying signal. In [20], the authors propose a solution where they sample at two distinct frequencies, f_1 and f_2, such that $f_1 > f_2$ and $\frac{f_1}{f_2}$ is not an integer. If aliasing occurs, i.e., the underlying signal has frequency terms that are larger than $\frac{f_2}{2}$, then comparing the discrete fourier transforms of the two sampled signals would show discrepancies. A complicating aspect here is noise at higher frequencies, but noise especially of a small amplitude can be filtered [1, 3, 30].

Collecting samples at two frequencies roughly doubles measurement cost but we still expect sizable net benefit since, as we saw in Section 3 current systems, over-sample by well over 2×. Furthermore, after checking for aliasing, we can discard excess measurements by resampling at the identified Nyquist rate. We believe that further improvements are possible for example by using an aliasing detector that is specific to changes that appear in data center measurements.

4.2 Adapting the sampling rate

Upon detecting aliasing, we increase the sampling rate. There are several possible approaches to manage the rate adaptation process and the choice among these depends on the properties of the signal (e.g., whether the high-frequency changes are one-off occurrences or sustained shifts) and the requirements for the measurement (e.g., how important is it to capture every spike and how costly is it to oversample).

Consider the problem of quantifying link failures by sampling frame checksum [37] errors. Initially, we do not know the Nyquist rate of the underlying signal and so we must probe: multiplicatively increase the measurement rate along with the method in Section 4.1, (i.e., measuring at two different frequencies) to detect aliasing. While aliasing persists, we remain in probe mode. Once we no longer detect aliasing, we use the method in Section 3.2 which will successfully identify the Nyquist rate of the signal: our new sampling rate. We continue to use the aliasing detection method to detect if aliasing recurs and if so move to probing mode again.

We can add memory into the system and/or leverage temporal stability to make adaptation faster. If the frequency increases exhibit temporal locality (as it does in, e.g., fail-stop or link-flap scenarios), we can optimize the system by also adaptively decreasing the sampling rate if we observe the Nyquist rate returning to a lower value. We can even 'remember' previous maximum Nyquist rates to ramp up more quickly in the future. Similarly, we may be able to learn information about signals' Nyquist shift distributions from other (oversampled) sources measuring the same signal. The optimal strategy will vary depending on the signal.

Perhaps the most challenging scenario for such a system is how the system should handle a first-of-its-kind event. Maintaining ample headroom may be helpful in these cases (many of the deployed systems we examine in Section 3 are already sampling at rates well above the Nyquist rate). We note, however, the most critical issues in modern systems are those that recur—after all, ongoing fires are typically a higher priority for operators to debug than events that occur only once and never again. An adaptive strategy will detect this more severe category of issues.

4.3 Quantization Noise and Reconstruction

In practice, measurements are quantized. For example, a temperature sensor may emit readings that are rounded to the nearest integer. Such quantization adds noise which in the frequency domain appears at higher frequencies; the larger the quanta relative to the range of values a signal can take, the higher the noise level. Quantization noise impacts our techniques in a few different ways: (a) identifying the true Nyquist rate of a signal becomes more challenging; and (b) upsampling and recovering the signal after it is downsampled. For (a) we use the thresholding approach proposed in Section 3.2 so as to discard higher-order frequencies introduced by quantization. For (b) we can add the same quantization in order to recover the signal more accurately. However, in such cases the signal is no longer "perfectly recoverable" and the recovered signal may be slightly different from the original. We show the effectiveness of this approach in [31].

Using the Nyquist principle and the adaptive sampling approach described above, we are able to reduce the overhead on the monitoring system. To reconstruct the signal, operators would have to pass the signal through a low-pass filter. This reconstruction takes time and may not be acceptable to applications that expect low-latency. However, in many cases this reconstruction cost is acceptable. For instance, machine learning models would typically prefer some delay in recovering the data in return for higher-fidelity.

5 RELATED WORK

In this paper, we proposed a different perspective on how we approach network monitoring. The research we propose builds on top of the vast array of knowledge in the field of signal processing and relates to the following categories of work in signal processing and network monitoring:

Signal processing theory and its applications to networking. The Nyquist–Shannon theorem is ubiquitous and has been applied in several domains [7, 22, 34] and been expanded in its applicability [23]. We build on this work.

The most closely related work are the existing, but isolated examples of signal processing techniques in systems and networks. These include techniques like compressive sensing. In [12], for example, authors regard flow statistics as signals to design new sketch algorithms that bound the information loss of the system. In [36] the authors use FFTs to identify network faults. Other work, characterize the underlying properties of network traffic [4, 6, 19, 24, 29]. These examples are complementary to our work.

Network monitoring systems. Many others have also tried to address the problem of scalability/overhead in both the collection and storage of network measurement. Sketches like the one in [12] and [13, 15] are one such approach. Sketches reduce the space required for storing data plane statistics, but those statistics must still be sampled when summarized for users or stored for later analysis.

There is also a vast body of network monitoring systems, from simple, periodic ones [9, 37], to those that are further optimized [11, 14, 16, 25]. It may be possible to improve the efficiency of these systems by re-framing the monitored information as time-varying signals as demonstrated by [12].

Many network protocols continuously measure (most prominently TCP) the network. Here too we may be able to improve efficiency through our new lens of viewing the information they monitor as a continuous or discrete-time signals.

In summary, this paper advocates for a broader re-examination of modern data center systems with an eye toward directly improving the efficiency of those systems by bringing to bear signal processing techniques.

6 CONCLUSION AND FUTURE WORK

Monitoring systems are a crucial part of ensuring high availability for data centers. Yet, efficient measurements based on information theoretic concepts are not widely studied. We argue the next stage of network monitoring should go beyond vague 'gut feelings' about the granularity of data and instead leverage concepts from signal processing to help avoid wasteful collection. We show how Nyquist-Shannon might provide guidance to optimize existing monitoring systems and propose a straw-man approach for adaptive sampling.

Beyond numbers. In this paper we focused on monitoring systems that fit nicely into the Nyquist model of sampling: they periodically sample a numeric metric at a fixed frequency. We need further research to apply these techniques to other more complex systems such as [12, 14–16, 33], which may measure events, sets of metrics, or text representations.

Multivariate signals. Many applications may monitor and use multiple different signals. The joint distribution of these signals may be important to such applications. As long as we sample each individual signal at a rate higher than Nyquist, we can recover the original signal. However, if the Nyquist rate changes frequently and the system attempts to use the dynamic increase/decrease approach of Section 4, the algorithm we presented may encounter pathological cases. For example, before the sampling rate of individual signals converges to the Nyquist rate, we may lose correlation between signals. Luckily, a number of work extend the Nyquist theorem to multivariate signals [8, 17]. Incorporating these extensions into a practical system is left to future work.

Beyond Nyquist. The Nyquist–Shannon theorem is one example of a technique from the realm of signal processing that can be applied to data centers. Another is ergodicity.

An ergotic process is one where the statistical properties of a sufficiently long random sample of the process are equivalent to the properties of a random sample of a statistical ensemble of the process. For example, consider a system that monitors the CPU utilization of all servers in a data center. Samples from the system are ergotic if the statistical properties of a set of samples derived from a single CPU over a sufficiently long sequence of time are equivalent to those of a set of samples from measuring the entire fleet at once.

Operators often assume ergodicity implicitly. One example is the practice of canarying, where an update is rolled out to a handful of servers/racks/switches to evaluate its effects before deploying the update more broadly. Extrapolating canary results to other devices relies on ergodicity. Does this assumption hold in practice? How long of an observation period is required for the assumption to hold? Is there a way to leverage ergodicity to reduce the number of devices that we need to sample?

REFERENCES

[1] Mariam Al Harrach, Sofiane Boudaoud, Mahmoud Hassan, Fouaz Sofiane Ayachi, D Gamet, Jean-François Grosset, and Frédéric Marin. 2017. Denoising of HD-sEMG signals using canonical correlation analysis. *Medical & biological engineering & computing* (2017).

[2] Behnaz Arzani, Selim Ciraci, Stefan Saroiu, Alec Wolman, Jack Stokes, Geoff Outhred, and Lechao Diwu. 2020. PrivateEye: Scalable and Privacy-Preserving Compromise Detection in the Cloud. In *USENIX NSDI*.

[3] Fábio M Bayer, Alice J Kozakevicius, and Renato J Cintra. 2019. An iterative wavelet threshold for signal denoising. *Signal Processing* (2019).

[4] Theophilus Benson, Aditya Akella, and David A. Maltz. 2010. Network Traffic Characteristics of Data Centers in the Wild. In *ACM IMC*. Association for Computing Machinery.

[5] Benoit Claise, Ganesh Sadasivan, Vamsi Valluri, and Martin Djernaes. 2004. Cisco systems netflow services export version 9. (2004).

[6] Erik D. Demaine, Alejandro López-Ortiz, and J. Ian Munro. 2002. Frequency Estimation of Internet Packet Streams with Limited Space. In *Algorithms — ESA 2002*, Rolf Möhring and Rajeev Raman (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg.

[7] Christopher L Farrow, Margaret Shaw, Hyunjeong Kim, Pavol Juhás, and Simon JL Billinge. 2011. Nyquist-Shannon sampling theorem applied to refinements of the atomic pair distribution function. *Physical Review B* (2011).

[8] Willi Freeden and M Zuhair Nashed. 2018. From the Gaussian Circle Problem to Multivariate Shannon Sampling. In *Frontiers in Orthogonal Polynomials and q-Series*. World Scientific.

[9] Chuanxiong Guo, Lihua Yuan, Dong Xiang, Yingnong Dang, Ray Huang, Dave Maltz, Zhaoyi Liu, Vin Wang, Bin Pang, Hua Chen, Zhi-Wei Lin, and Varugis Kurien. 2015. Pingmesh: A Large-Scale System for Data Center Network Latency Measurement and Analysis. In *ACM SIGCOMM*.

[10] Chris Hare. 2011. Simple Network Management Protocol (SNMP).

[11] Thomas Holterbach, Edgar Costa Molero, Maria Apostolaki, Alberto Dainotti, Stefano Vissicchio, and Laurent Vanbever. 2019. Blink: Fast connectivity recovery entirely in the data plane. In *USENIX NSDI*.

[12] Qun Huang, Siyuan Sheng, Xiang Chen, Yungang Bao, Rui Zhang, Yanwei Xu, and Gong Zhang. 2021. Toward Nearly-Zero-Error Sketching via Compressive Sensing. In *USENIX NSDI*. USENIX Association.

[13] Nikita Ivkin, Zhuolong Yu, Vladimir Braverman, and Xin Jin. 2019. Qpipe: Quantiles sketch fully in the data plane. In *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*.

[14] Masoud Moshref, Minlan Yu, Ramesh Govindan, and Amin Vahdat. 2014. Dream: dynamic resource allocation for software-defined measurement. In *ACM SIGCOMM*.

[15] Masoud Moshref, Minlan Yu, Ramesh Govindan, and Amin Vahdat. 2015. Scream: Sketch resource allocation for software-defined measurement. In *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies*.

[16] Masoud Moshref, Minlan Yu, Ramesh Govindan, and Amin Vahdat. 2016. Trumpet: Timely and precise triggers in data centers. In *ACM SIGCOMM*.

[17] Amos Nathan. 1978. Plain and covariant multivariate fourier transforms. *Information and Control* (1978).

[18] Alan V Oppenheim and Ronald W Schafer. 1975. Digital signal processing(Book). *Research supported by the Massachusetts Institute of Technology, Bell Telephone Laboratories, and Guggenheim Foundation. Englewood Cliffs, N. J., Prentice-Hall, Inc., 1975. 598 p* (1975).

[19] Philippe Owezarski and Nicolas Larrieu. 2004. Internet Traffic Characterization – An Analysis of Traffic Oscillations. In *High Speed Networks and Multimedia Communications*, Zoubir Mammeri and Pascal Lorenz (Eds.). Springer Berlin Heidelberg.

[20] J. E. T. Penny, M. I. Friswell, and S. D. Garvey. [n.d.]. Detecting Aliased Frequency Components in Discrete Fourier Transforms. *Mechanical Systems and Signal Processing* ([n. d.]).

[21] Peter Phaal, Sonia Panchen, and Neil McKee. 2001. RFC3176: InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks.

[22] G Romo-Cárdenas, GJ Avilés-Rodríguez, J de D Sánchez-López, M Cosío-León, PA Luque, CM Gómez-Gutiérrez, Juan I Nieto-Hipólito, Mabel Vázquez-Briseño, and Christian X Navarro-Cota. 2018. Nyquist-Shannon theorem application for Savitzky-Golay smoothing window size parameter determination in bio-optical signals. *Results in Physics* (2018).

[23] Zhanjie Song, Bei Liu, Yanwei Pang, Chunping Hou, and Xuelong Li. 2012. An improved Nyquist–Shannon irregular sampling theorem from local averages. *IEEE transactions on information theory* (2012).

[24] Stilian Stoev, Murad S. Taqqu, Cheolwoo Park, and J.S. Marron. 2005. On the wavelet spectrum diagnostic for Hurst parameter estimation in the analysis of Internet traffic. *Computer Networks* (2005).

[25] Praveen Tammana, Rachit Agarwal, and Myungjin Lee. 2016. Simplifying datacenter network debugging with pathdump. In *USENIX OSDI*.

[26] Lizhuang Tan, Wei Su, Wei Zhang, Jianhui Lv, Zhenyi Zhang, Jingying Miao, Xiaoxi Liu, and Na Li. 2021. In-band network telemetry: a survey. *Computer Networks* (2021).

[27] Philippe Thévenaz, Thierry Blu, and Michael Unser. 2000. Image interpolation and resampling. *Handbook of medical imaging, processing and analysis* (2000).

[28] P. Thévenaz, T. Blu, and M. Unser. 2000. Image Interpolation and Resampling. In *Handbook of Medical Imaging, Processing and Analysis*. Academic Press.

[29] K. Thompson, G.J. Miller, and R. Wilder. 1997. Wide-area Internet traffic patterns and characteristics. *IEEE Network* (1997). https://doi.org/10.1109/65.642356

[30] Saeed V Vaseghi. 2008. *Advanced digital signal processing and noise reduction*. John Wiley & Sons.

[31] Nofel Yaseen, Behnaz Arzani, Krishna Chintalapudi, Vaishnavi Ranganathan, Felipe Frujeri, Kevin Hsieh, Daniel Berger, Vincent Liu, and Srikanth Kandula. 2021. Towards a Cost vs. Quality Sweet Spot for Monitoring Networks. *arXiv preprint* (2021).

[32] Nofel Yaseen, John Sonchack, and Vincent Liu. 2018. Synchronized Network Snapshots. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication (ACM SIGCOMM)*.

[33] Nofel Yaseen, John Sonchack, and Vincent Liu. 2020. tpprof: A Network Traffic Pattern Profiler. In *USENIX NSDI*. USENIX Association.

[34] VV Zamaruiev. 2017. The use of Kotelnikov-Nyquist-Shannon sampling theorem for designing of digital control system for a power converter. In *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*. IEEE.

[35] Qiao Zhang, Vincent Liu, Hongyi Zeng, and Arvind Krishnamurthy. 2017. High-Resolution Measurement of Data Center Microbursts *(ACM IMC)*. Association for Computing Machinery.

[36] Yin Zhang, Zihui Ge, Albert Greenberg, and Matthew Roughan. 2005. Network anomography. In *ACM IMC*.

[37] Danyang Zhuo, Monia Ghobadi, Ratul Mahajan, Klaus-Tycho Förster, Arvind Krishnamurthy, and Thomas Anderson. 2017. Understanding and mitigating packet corruption in data center networks. In *ACM SIGCOMM*.