How Accurate is Selective INT for Traffic Trace Reconstruction and How to Adjust it Adaptively?

Zichen Xu and Zuqing Zhu
School of Information Science and Technology, University of Science and Technology of China, Hefei, China
Email:{zqzhu}@ieee.org

***Abstract*—In-band network telemetry (INT) can monitor networks in a flow-oriented and realtime way. The bandwidth overheads caused by per-packet INT can be effectively reduced by selective INT (Sel-INT), *i.e.*, sampling packets or/and types of telemetry data for INT. For the sampling in Sel-INT, there is a tradeoff between bandwidth overheads and monitoring accuracy, which can be adjusted by the sampling ratio. In this paper, we investigate how to adjust the sampling ratio of Sel-INT adaptively such that the tradeoff can be balanced precisely. We first develop a theoretical model to analyze the accuracy of Sel-INT for traffic trace reconstruction under different sampling ratios. We assume that the data samples of a flow's traffic trace (*i.e.*, bandwidth usage) follow the Gaussian random process, and the traffic trace is reconstructed with linear interpolation based on the samples collected by Sel-INT. We analyze this process theoretically and derive the expected reconstruction error (RCE) between the original and reconstructed traces. Then, we use RCE to determine whether the sampling ratio of Sel-INT is properly set and propose two algorithms, namely, RCESA and pRCESA, to adjust the sampling ratio adaptively. Extensive simulations with realistic traffic traces verify the effectiveness of our proposal.**

***Index Terms*—In-band network telemetry (INT), Selective INT (Sel-INT), Sampling error, Traffic monitoring.**

I. Introduction

Recent advances on data-centers and 5G networks have greatly reshaped Internet infrastructures [1] and stimulated innovations in a number of areas (*e.g.*, physical-layer technologies [2–5] and network virtualization [6–9]). Despite their advantages, these innovations make networks more prone to faults and thus complicate network monitoring and troubleshooting. Therefore, today's network operators are eager to have network monitoring techniques that can visualize network status in a fine-grained, realtime, and efficient manner, such that exceptions can be detected, located, and recovered quickly. However, these requirements can hardly be satisfied by the conventional network monitoring techniques [10], because they either are not precise enough to capture network status in realtime or cannot reveal the end-to-end operation of flows.

This dilemma can be overcome by leveraging the progresses on programmable data plane (PDP) [11, 12] to create novel network monitoring techniques. For instance, in-band network telemetry (INT) [13] has been proposed to monitor networks and locate exceptions in a flow-oriented, realtime, and effective way. Implemented with PDP, INT inserts an INT header in each packet of a service flow. The INT header consists of

978-1-6654-3540-6/22 © 2022 IEEE

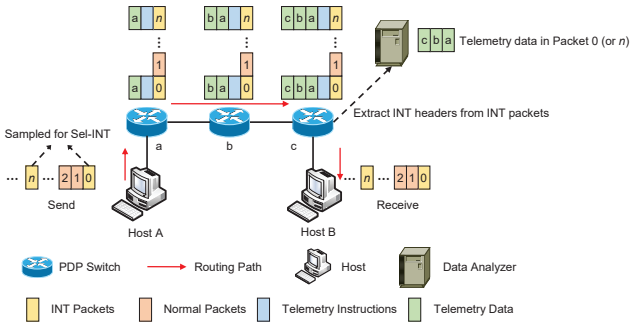


Fig. 1. Operation principle of Sel-INT.

specific INT fields to store telemetry instructions for the flow and the telemetry data that is collected on each hop according to the instructions. Hence, by extracting the INT header at the flow's egress switch, one can analyze the telemetry data in it to reveal how the packet was processed in a network.

Note that, INT samples the network status experienced by a flow in a per-packet way. This might not be necessary since the operation of a network usually would not change dramatically at the time granularity of a single packet, especially when the flow's data-rate is relatively high. Therefore, people have proposed several schemes to reduce the bandwidth overheads of INT and avoid seizing too much bandwidth from normal traffic, by sampling packets or/and types of telemetry data for INT [14–17]. These schemes categorize packets in a flow into two types: 1) the **normal packets** that are not selected for INT, and 2) the **INT packets** that carry INT headers. For instance, Fig. 1 explains the operation of the selective INT (Sel-INT) in [16] when the sampling ratio of INT packets is $\frac{1}{n}$.

When sampling is considered for INT, there is a tradeoff between bandwidth overheads and monitoring accuracy, which can be adjusted by the sampling ratio. Although previous studies have considered the tradeoff and explained how to adjust it in different scenarios [16–18], none of them has tackled the problem from the theoretical perspective. Hence, their approaches for adjusting the sampling ratio could be arbitrary and might not always balance the tradeoff properly.

In this work, we first develop a theoretical model to analyze the accuracy of Sel-INT for traffic trace reconstruction under different sampling ratios. We assume that the data samples of a flow's traffic trace (*i.e.*, bandwidth usage) follow the Gaussian random process, and the traffic trace is first collected with Sel-INT and then reconstructed with linear interpolation based on the sampled results. We analyze the process theoretically and

derive the expected reconstruction error (RCE) between the original and reconstructed traces based on conditional mean and variance (CMV). Next, as RCE can quantify the accuracy of traffic trace reconstruction precisely, we use it to determine whether the sampling ratio of Sel-INT is properly set and propose two algorithms to adjust the sampling ratio adaptively. Finally, we conduct extensive simulations to demonstrate the effectiveness of our proposed model and algorithms.

The rest of this paper is organized as follows. Section II surveys the related work briefly. In Section III, we describe our problem and derive the theoretical model for RCE. The algorithms are designed in Section IV and the simulations are shown in Section V. Finally, Section VI summarizes the paper.

II. Related Work

Since INT provides a promising way to monitor networks effectively, it has attracted intensive R&D interests. The first specification of INT has been released in [13], which suggests to collect telemetry data on a per-packet basis. To reduce the redundant bandwidth overheads of per-packet INT, previous studies [14–17] proposed a few schemes to sample packets or/and types of telemetry data for INT. The studies in [14, 15] assumed a preset sampling ratio, Sel-INT in [16] can be programmed in runtime to change the sampling ratio on-the-fly, and PINT in [17] leveraged probabilistic sampling to select the types of telemetry data for INT. Nevertheless, none of these sampling schemes tried to determine the sampling ratio with a solid theoretical model, and thus their ways of selecting the sampling ratio could be arbitrary and might not always balance the tradeoff between the bandwidth overheads and monitoring accuracy of INT properly.

In addition to sampling schemes, people have also considered to reduce the bandwidth overheads of INT by planning flow paths to avoid collecting redundant telemetry data and to maximize the coverage of network monitoring (*e.g.*, in [19]). However, these proposals cannot reduce the bandwidth overheads of INT on each flow after its path has been determined.

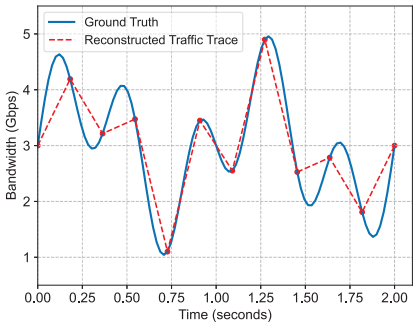
To the best of our knowledge, none of the existing studies has analyzed the accuracy of INT sampling schemes theoretically or proposed algorithms to adjust the sampling ratio precisely to balance the tradeoff between the bandwidth overheads and monitoring accuracy of INT properly.

III. Theoretical Analysis

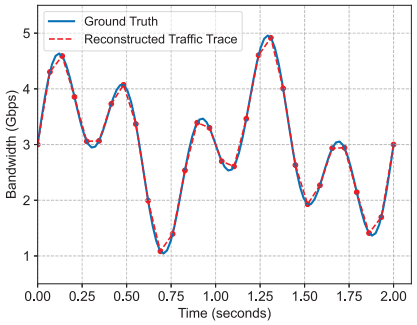
In this section, we first present the problem of analyzing the accuracy of Sel-INT for traffic trace reconstruction, and then derive the theoretical model to solve it.

A. Problem Description

Per-packet INT collects the network status experienced by a flow with the highest accuracy, and thus we can treat its monitoring results as the "ground truth". On the other hand, Sel-INT samples the network status with a sampling ratio to reduce the bandwidth overheads of network monitoring, and then relies on the data analyzer (DA) to reconstruct the original network status with the sampled results (as shown in Fig. 1). Although Sel-INT can effectively reduce the bandwidth overheads, it could bring in RCE and make the reconstructed network status be different from the ground truth, especially when the sampling ratio is not properly set.



(a) Reconstruction in case of under-sampling



(b) Reconstruction in case of proper sampling

Fig. 2. Examples of traffic trace reconstruction with Sel-INT.

Note that, Sel-INT can collect various types of telemetry data, which can be categorized as static ones (*e.g.*, *In Port/Out Port* and *Device ID*), slow-varying ones (*e.g.*, *Hop Latency*), and fast-varying ones (*e.g.*, *Bandwidth*) [16]. As the reconstruction of static and slow-varying telemetry data will not be affected significantly with the sampling ratio, we only study the RCE of the fast-varying telemetry data (*i.e.*, *Bandwidth*) in this work. Specifically, we assume that Sel-INT samples bandwidth usage of a flow with a preset sampling ratio and then reconstructs the original bandwidth usage with linear interpolation, and develop a theoretical model to analyze the RCE generated in the process. Fig. 2 gives examples on traffic trace reconstruction with Sel-INT. When the sampling ratio is too small, the reconstructed trace in Fig. 2(a) does not match with the ground truth, but if the sampling ratio is properly set, that in Fig. 2(b) can approximate the ground truth well.

With a theoretical model to estimate the RCE, we can set an upper-limit on it to ensure the accuracy of traffic trace reconstruction, and then adjust the sampling ratio of Sel-INT accordingly to balance the tradeoff between bandwidth overheads and monitoring accuracy precisely.

B. Theoretical Model

The bandwidth usage of a dynamic flow can be highly uncertain, and people have tried to model it with various stochastic processes. Among them, the Gaussian process has been used to model traffic traces in different networks [20–22], and more importantly, it enables us to obtain closed-form

expressions in theoretical analysis. Therefore, we leverage it in this work to model the traffic samples of a flow. Specifically, we assume that the traffic samples (*i.e.*, the ground truth) can be modeled with a stationary Gaussian random process, *i.e.*, each sample follows a Gaussian distribution whose mean and variance are μ and σ^2, respectively. Here, the values of μ and σ^2 are known (*e.g.*, obtained by historical observations).

We denote the traffic samples of the ground truth as $\{x(t_1), \cdots, x(t_n)\}$, where $\{t_1, \cdots, t_n\}$ are the indices of the time instances when the samples could be collected with per-packet INT. Similarly, the reconstructed samples with Sel-INT can be denoted as $\{\widetilde{x}(t_1), \cdots, \widetilde{x}(t_n)\}$. Then, the error at an arbitrary time index t_i can be obtained as $|x(t_i) - \widetilde{x}(t_i)|$.

Definition 1: We define the **reconstruction error (RCE)** of n consecutive samples reconstructed by Sel-INT as

$$\xi_{\text{RCE}}(n) = \frac{1}{n} \sum_{i=1}^{n} \left[\widetilde{x}(t_i) - x(t_i)\right]^2. \tag{1}$$

We assume that the indices of the time instances for Sel-INT to sample the traffic trace are $\{T_1, \cdots, T_N\}$. Then, the traffic samples collected with Sel-INT are $\{x(T_1), \cdots, x(T_N)\}$. Then, for any two consecutive samples $x(T_i)$ and $x(T_{i+1})$, we use linear interpolation to reconstruct the skipped traffic samples $\{x(t_1^*), \cdots, x(t_m^*)\}$ between them

$$\widetilde{x}(t_k^*) = x(T_i) + \frac{k}{m+1} \cdot [x(T_{i+1}) - x(T_i)]. \tag{2}$$

Here, the sampling ratio of Sel-INT is $\frac{1}{m+1}$. As the actual values of the skipped traffic samples $\{x(t_1^*), \cdots, x(t_m^*)\}$ are unknown, we have to treat them as random variables. Therefore, the expectation of the RCE of $\{\widetilde{x}(t_1^*), \cdots, \widetilde{x}(t_m^*)\}$ is

$$\begin{aligned} \mathbb{E}(\xi_{\text{RCE}}(m)) &= \mathbb{E}\left(\frac{1}{m}\sum_{k=1}^{m}\left[\widetilde{x}(t_k^*) - x(t_k^*)\right]^2\right) \\ &= \frac{1}{m}\sum_{k=1}^{m}\left[\mathbb{E}(x^2(t_k^*)) - 2\cdot\widetilde{x}(t_k^*)\cdot\mathbb{E}(x(t_k^*)) + \widetilde{x}^2(t_k^*)\right]. \end{aligned} \tag{3}$$

It is known that the traffic samples of a network flow are long-range dependent [23, 24]. Hence, the values of $\mathbb{E}(x^2(t_k^*))$ and $\mathbb{E}(x(t_k^*))$ should be dependent on the collected samples $x(T_i)$ and $x(T_{i+1})$. We define the conditional mean $\mu(t_k^*) = \mathbb{E}(t_k^*|x(T_i), x(T_{i+1}))$ and conditional variance $\sigma^2(t_k^*) = \mathbb{V}(t_k^*|x(T_i), x(T_{i+1}))$ and calculate them as

$$\begin{cases} \mathbb{E}(x(t_k^*)) = \mu(t_k^*), \\ \mathbb{E}(x^2(t_k^*)) = \mu^2(t_k^*) + \sigma^2(t_k^*), \end{cases} \tag{4}$$

Then, the expectation of RCE in Eq. (3) gets transformed into

$$\mathbb{E}(\xi_{\text{RCE}}(m)) = \frac{1}{m}\sum_{k=1}^{m}\left[\mu^2(t_k^*) + \sigma^2(t_k^*) + \widetilde{x}^2(t_k^*) - 2\widetilde{x}(t_k^*)\mu(t_k^*)\right]. \tag{5}$$

We can derive the values of $\mu(t_k^*)$ and $\sigma^2(t_k^*)$ as [25]

$$\mu(t_k^*) = \mu + \sum_{i'=1}^{2}\sum_{j'=1}^{2}\mathbb{K}(t_k^*, T_{i+i'-1})\cdot a_{i',j'}\cdot[x(T_{i+j'-1}) - \mu], \tag{6}$$

$$\sigma^2(t_k^*) = \sigma^2 - \sum_{i'=1}^{2}\sum_{j'=1}^{2}\mathbb{K}(t_k^*, T_{i+i'-1})\cdot a_{i',j'}\cdot\mathbb{K}(T_{i+j'-1}, t_k^*). \tag{7}$$

Here, $\mathbb{K}(t_i, t_j)$ is the covariance function of traffic samples $x(t_i)$ and $x(t_j)$. As we assume that the traffic samples follow a stationary random process, the value of $\mathbb{K}(t_i, t_j)$ only depends on the index interval $\tau = |t_i - t_j|$. Hence, we have $\mathbb{K}(t_i, t_j) = \mathbb{K}(\tau)$ and can estimate its value from historical traffic samples of the trace. Specifically, if we assume that a series of N samples $\{y_1, \cdots, y_N\}$ have already been collected for the flow, the covariance function can be estimated as

$$\mathbb{K}(t_i, t_j) = \mathbb{K}(\tau) = \frac{1}{N-\tau}\sum_{i=1}^{N-\tau}(y_i - \mu)\cdot(y_{i+\tau} - \mu). \tag{8}$$

Meanwhile, the $a_{i',j'}$ in Eqs. (6) and (7) refers to a corresponding element in the inverse covariance matrix $\boldsymbol{A}$:

$$\boldsymbol{A} = \begin{bmatrix} \mathbb{K}(T_i, T_i) & \mathbb{K}(T_i, T_{i+1}) \\ \mathbb{K}(T_{i+1}, T_i) & \mathbb{K}(T_{i+1}, T_{i+1}) \end{bmatrix}^{-1} = \begin{bmatrix} \sigma^2 & \mathbb{K}(\tilde{\tau}) \\ \mathbb{K}(\tilde{\tau}) & \sigma^2 \end{bmatrix}^{-1}, \tag{9}$$

where $\tilde{\tau} = T_{i+1} - T_i$ denotes the sampling period of Sel-INT, and $\mathbb{K}(\tilde{\tau})$ can be obtained with Eq. (8).

Theorem 1: The matrix $\boldsymbol{A}$ defined in Eq. (9) always exists.

Proof: The inverse covariance matrix $\boldsymbol{A}$ should satisfy

$$\boldsymbol{A}\cdot\begin{bmatrix} \sigma^2 & \mathbb{K}(\tilde{\tau}) \\ \mathbb{K}(\tilde{\tau}) & \sigma^2 \end{bmatrix} = \begin{bmatrix} \sigma^2 & \mathbb{K}(\tilde{\tau}) \\ \mathbb{K}(\tilde{\tau}) & \sigma^2 \end{bmatrix}\cdot\boldsymbol{A} = \boldsymbol{I}, \tag{10}$$

where $\boldsymbol{I}$ is a 2×2 identity matrix. Hence, we can get

$$\boldsymbol{A} = \begin{bmatrix} \frac{\sigma^2}{\sigma^4 - \mathbb{K}^2(\tilde{\tau})} & \frac{\mathbb{K}(\tilde{\tau})}{\mathbb{K}^2(\tilde{\tau}) - \sigma^4} \\ \frac{\mathbb{K}(\tilde{\tau})}{\mathbb{K}^2(\tilde{\tau}) - \sigma^4} & \frac{\sigma^2}{\sigma^4 - \mathbb{K}^2(\tilde{\tau})} \end{bmatrix}. \tag{11}$$

It can be seen that $\boldsymbol{A}$ will not exist only when we have $\mathbb{K}(\tilde{\tau}) = \sigma^2$. However, by definition, we only have $\mathbb{K}(\tilde{\tau}) = \sigma^2$ when $\tilde{\tau} = 0$. In other words, $\mathbb{K}(\tilde{\tau})$ is a monotonically decreasing function and its maximum is σ^2 when $T_{i+1} - T_i = 0$. Note that, the operation principle of Sel-INT ensures that T_i and T_{i+1} are for different time instants. Hence, the matrix $\boldsymbol{A}$ always exists, and it can be calculated with Eq. (11). ■

In all, with Eqs. (2), (5)-(8), and (11), we can get the expected RCE of the reconstructed samples $\{\widetilde{x}(t_1^*), \cdots, \widetilde{x}(t_m^*)\}$ between $x(T_i)$ and $x(T_{i+1})$, and can use it to determine whether the current sampling ratio of Sel-INT is proper set to ensure the accuracy of traffic trace reconstruction.

IV. Algorithm Design

In this section, we propose two algorithms to adjust the sampling ratio of Sel-INT adaptively according to the expectation of RCE and an upper-limit on RCE.

A. RCE-Sensitive Algorithm (RCESA)

With an upper-limit on RCE to ensure the accuracy of traffic trace reconstruction, we first design a RCE-sensitive algorithm (RCESA) to adjust the sampling ratio of Sel-INT adaptively. *Algorithm* 1 shows the procedure of RCESA. As for the inputs of *Algorithm* 1, $\hat{\xi}$ denotes the upper-limit of RCE for ensuring the accuracy of traffic trace reconstruction, m is the initial sampling interval (*i.e.*, the number of traffic samples that Sel-INT skips between two consecutive data collections initially), $\eta \in (0, 1)$ is the proportional coefficient that will be applied

to $\hat{\xi}$, and N is the threshold for the collections conducted by Sel-INT, above which the sampling ratio can be changed.

Specifically, the rationale of introducing η and N is as follows. As we try to maintain the expected RCE below $\hat{\xi}$, the sampling ratio of Sel-INT should be increased immediately if we observe $\mathbb{E}(\xi_{\text{RCE}}) \geq \hat{\xi}$. Otherwise, when we have $\mathbb{E}(\xi_{\text{RCE}}) < \hat{\xi}$, we should be cautious about decreasing the sampling ratio. This is because decreasing the sampling ratio too fast can make the expected RCE exceed $\hat{\xi}$ and thus invoke unnecessary adjustments. Hence, we introduce η and N to ensure that the sampling ratio is only reduced when necessary.

In *Algorithm* 1, *Line* 1 is for the initialization. Here, we allocate a variable x to store the last traffic sample collected by Sel-INT, and it is initialized as -1 to avoid confusion (*i.e.*, a valid traffic sample should be non-negative). k_1 and k_2 are two counters to determine whether the sampling ratio should be decreased. After getting a new traffic sample with Sel-INT, *Lines* 3-20 update the sampling ratio adaptively. *Line* 4 calculates the expected RCE of the traffic trace reconstruction that is based on the latest collected sample with the theoretical model derived in the previous section.

If the expected RCE reaches the preset upper-limit $\hat{\xi}$, *Lines* 5-6 decrease the sampling interval m immediately and reset k_1 and k_2. Otherwise, *Lines* 7-18 adjust the sampling ratio according to the actual value of $\mathbb{E}(\xi_{\text{RCE}})$. If we have $\mathbb{E}(\xi_{\text{RCE}}) < \eta \cdot \hat{\xi}$, which means that the expected RCE has certain distance from its upper-limit, we will increase the sampling interval m after the situation has been persisted for N samples from Sel-INT (*Lines* 7-15). Otherwise, we have $\mathbb{E}(\xi_{\text{RCE}}) \in [\eta \cdot \hat{\xi}, \hat{\xi})$, which means that even though the expected RCE is smaller than its upper-limit, the gap between them is still relatively small. Hence, we will only flag the case by increasing k_2 but will not change the sampling ratio.

Algorithm 1: RCE-Sensitive Algorithm

```
   Input: upper-limit of RCE ξ̂, initial sampling interval m,
          proportional coefficient η, threshold on samples N
1  k_1 = k_2 = 0, x = −1;
2  while get a new traffic sample x(T_i) with Sel-INT do
3      if x ≠ −1 then
4          calculate E(ξ_RCE) based on x and x(T_i) with Eqs. (2),
           (5)-(8), and (11);
5          if E(ξ_RCE) ≥ ξ̂ then
6              m = m − 1, k_1 = k_2 = 0;
7          else if E(ξ_RCE) < η · ξ̂ then
8              if k_2 > 1 then
9                  k_1 = 1, k_2 = 0;
10             else
11                 k_1 = k_1 + 1, k_2 = 0;
12             end
13             if k_1 = N then
14                 m = m + 1, k_1 = k_2 = 0;
15             end
16         else
17             k_2 = k_2 + 1;
18         end
19     end
20     x = x(T_i);
21 end
```

B. Probabilistic RCE-Sensitive Algorithm (pRCESA)

Although RCESA can ensure the accuracy of traffic trace reconstruction of Sel-INT to the maximum extent, it becomes sensitive when the expected RCE exceeds its upper-limit. However, packet losses may happen in a network, which can make Sel-INT lose some of its collected traffic samples and thus decrease the sampling ratio unexpectedly, resulting the expected RCE to exceed its upper-limit. In this case, increasing the sampling ratio might not be necessary, because it is a false positive to see that Sel-INT is under-sampling the traffic trace.

To address the issue above, we propose a probabilistic RCE-sensitive algorithm (pRCESA) to make the adjustment of the sampling ratio less sensitive to the cases of $\mathbb{E}(\xi_{\text{RCE}}) \geq \hat{\xi}$. *Algorithm* 2 shows the procedure of pRCESA. This time, we introduce four counters $\{k_i,\ i \in [1,4]\}$ to determine whether the sampling ratio should be changed. pRCESA increases the sampling ratio mainly based on the value of k_1, as shown in *Lines* 5-11. The $f(k_1)$ in *Line* 6 is a monotonically increasing function whose output range is $[0,1]$. Hence, if $\mathbb{E}(\xi_{\text{RCE}}) \geq \hat{\xi}$ does not happen frequently (*i.e.*, it is caused by occasional packet losses), the value of k_1 will remain as small and thus the probability of increasing the sampling ratio will be low, and *vice versa* (as shown in *Lines* 7-11, where the $random(0,1)$ in *Line* 7 returns a random real number within $(0,1)$). k_3 is used to reset k_1 to avoid a long interval between two consecutive counts (*Lines* 13-16), while k_4 is introduced to reset k_2 for the similar reason (*Lines* 18-22). The remaining operations in pRCESA are similar to those in RCESA. Even though pRCESA might not always ensure the accuracy of traffic trace reconstruction, it can save more bandwidth overheads, especially when there are occasional packet losses.

V. Performance Evaluation

Our simulations select 10 traces from the data set in [26], which were for traffic traces collected from real-world networks. Each trace consists of more than 800 samples, each of which denotes the bandwidth usage collected by a packet. We assume the bandwidth overhead of Sel-INT on each INT packet is 4 Bytes. We first try to verify our theoretical model about RCE in Section III-B. Specifically, we change the sampling interval $m \in [1,5]$, calculate the expected RCE $\mathbb{E}(\xi_{\text{RCE}}(m))$ with our model, and compare it with the real value in Fig. 3. Here, all of the 10 traces are considered, and we average the results to get each data point in Fig. 3. It can be seen that the RCEs from our model match with the real ones very well, confirming the accuracy of our model.

Next, we compare our algorithms with three benchmarks, *i.e.*, the per-packet INT and Sel-INT schemes with sampling intervals $m = \{1,4\}$. Fig. 4 shows the results on average RCE and bandwidth overheads when we change the preset upper-limit on RCE ($\hat{\xi}$) and assume that there is no packet loss. Here, we still consider all the traces, and for pRCESA, we run 100 independent simulations for each trace to emulate the scenarios due to its probabilistic adjustment, and average the results to get each data point. Fig. 4(a) indicates that our proposed algorithms (RCESA and pRCESA) adjust the sampling

Algorithm 2: Probabilistic RCE-Sensitive Algorithm

Input: upper-limit of RCE $\hat{\xi}$, initial sampling interval m, proportional coefficient η, threshold on samples N

$k_1 = k_2 = k_3 = k_4 = 0$, $x = -1$;
while *get a new traffic sample* $x(T_i)$ *with Sel-INT* **do**
  if $x \neq -1$ **then**
    calculate $\mathbb{E}(\xi_{\text{RCE}})$ based on x and $x(T_i)$ with Eqs. (2), (5)-(8), and (11);
    if $\mathbb{E}(\xi_{RCE}) \geq \hat{\xi}$ **then**
      $k_1 = k_1 + 1$, $p = f(k_1)$;
      if $random(0, 1) < p$ **then**
        $m = m - 1$, $k_1 = k_2 = k_3 = k_4 = 0$;
      else
        $k_3 = 0$;
      end
    else
      $k_3 = k_3 + 1$;
      if $k_3 > 2$ **then**
        $k_1 = k_3 = 0$;
      end
      if $\mathbb{E}(\xi_{RCE}) < \eta \cdot \hat{\xi}$ **then**
        if $k_1 + k_4 > 1$ **then**
          $k_2 = 1$, $k_4 = 0$;
        else
          $k_2 = k_2 + 1$, $k_4 = 0$;
        end
        if $k_2 = N$ **then**
          $m = m + 1$, $k_1 = k_2 = k_3 = k_4 = 0$;
        end
      else
        $k_4 = k_4 + 1$;
      end
    end
  end
  $x = x(T_i)$;
end

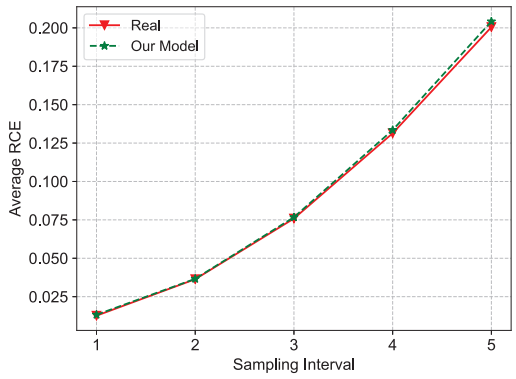
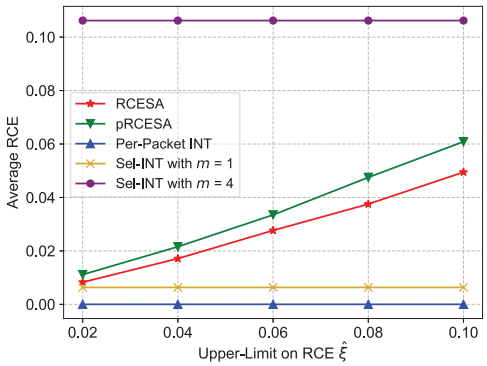
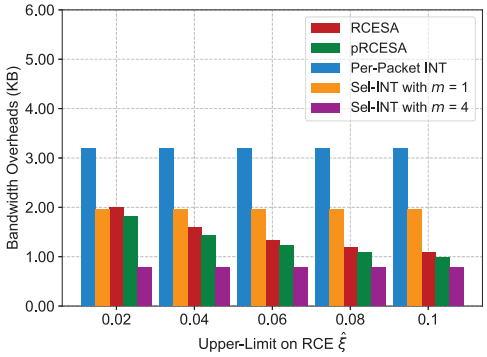


Fig. 3. Average RCE of Sel-INT with different sampling intervals.

interval of Sel-INT adaptively according to the requirement on RCE, while the benchmarks can only provide fixed RCE. The adaptivity of RCESA and pRCESA enables them to save bandwidth overheads intelligently, as shown in Fig. 4(b). Hence, the results in Fig. 4 confirm that our algorithms balance the tradeoff between bandwidth overheads and monitoring accuracy better than the benchmarks. Moreover, by comparing RCESA and pRCESA in Fig. 4, we find that pRCESA balances the aforementioned tradeoff better. Specifically, with a given upper-limit on RCE $\hat{\xi}$, RCESA can restrict RCE too small and thus introduce certain unnecessary bandwidth overheads.

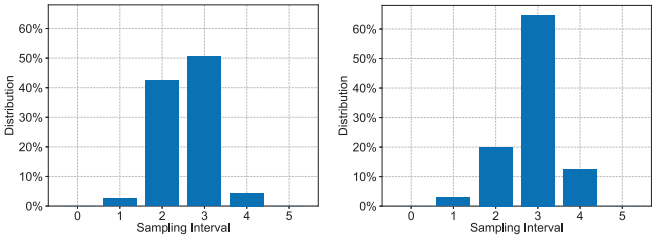


(a) RCE



(b) Bandwidth overheads

Fig. 4. Performance of algorithms without packet losses.



(a) RCESA (b) pRCESA

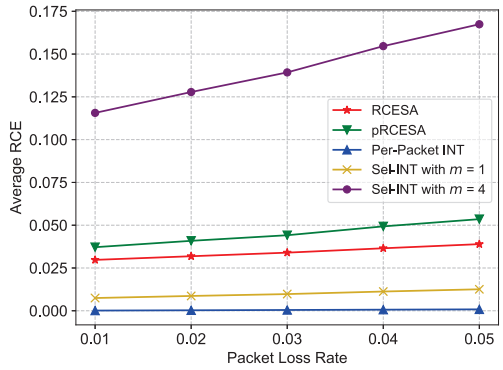
Fig. 5. Distribution of sampling intervals with $\hat{\xi} = 0.1$ and no packet loss.

The analysis can be verified with the distributions of sampling intervals in Fig. 5, which shows that pRCESA uses more large sampling intervals (*i.e.*, $m = \{3, 4\}$) than RCESA.

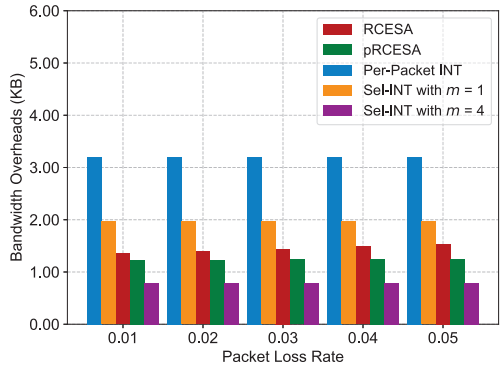
Finally, we consider the cases where packet losses are possible. This time, we fix $\hat{\xi} = 0.06$ and change the packet loss rate within $[0.01, 0.05]$. Fig. 6(a) shows that the RCEs of all the INT schemes increase with the packet loss rate, which is expected. The results in Fig. 6 still confirm that our algorithms balance the tradeoff between bandwidth overheads and monitoring accuracy better than the benchmarks. Meanwhile, pRCESA still performs better than RCESA, and the performance gap between them actually increases with the packet loss rate. Fig. 7 shows the distributions of the sampling intervals from RCESA and pRCESA, suggesting that pRCESA still tends to use more large sampling intervals than RCESA.

VI. Conclusion

In this paper, we studied the problem of how to adjust the sampling ratio of Sel-INT adaptively such that the tradeoff between bandwidth overheads and monitoring accuracy can

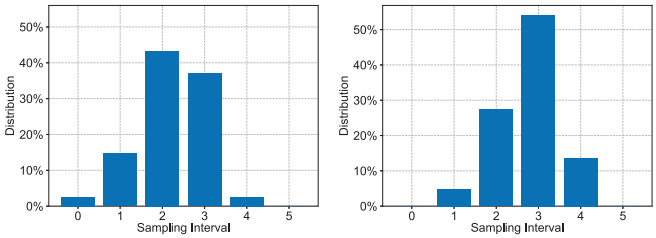


(a) RCE



(b) Bandwidth overheads

Fig. 6. Performance of algorithms with packet losses and $\hat{\xi} = 0.06$.



(a) RCESA (b) pRCESA

Fig. 7. Distribution of sampling intervals with packet loss rate as 0.05.

be balanced precisely. We first developed a theoretical model to analyze the RCE of Sel-INT for traffic trace reconstruction under different sampling ratios. Then, as RCE can quantify the accuracy of traffic trace reconstruction precisely, we leveraged it to determine whether the sampling ratio of Sel-INT is properly set and proposed two algorithms, namely, RCESA and pRCESA, to adjust the sampling ratio adaptively. We conducted extensive simulations with realistic traffic traces to evaluate our algorithms. The simulation results suggested that both RCESA and pRCESA can adjust the sampling interval of Sel-INT adaptively according to the requirement on RCE, and thus balance the tradeoff between bandwidth overheads and monitoring accuracy much better than existing benchmarks. Moreover, between RCESA and pRCESA, pRCESA performed better, especially when packet losses can happen.

Acknowledgments

This work was supported by NSFC project 61871357 and Fundamental Fund for Central Universities (WK3500000006).

References

[1] P. Lu *et al.*, "Highly-efficient data migration and backup for Big Data applications in elastic optical inter-datacenter networks," *IEEE Netw.*, vol. 29, pp. 36–42, Sept./Oct. 2015.

[2] Z. Zhu, W. Lu, L. Zhang, and N. Ansari, "Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing," *J. Lightw. Technol.*, vol. 31, pp. 15–22, Jan. 2013.

[3] L. Gong *et al.*, "Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks," *J. Opt. Commun. Netw.*, vol. 5, pp. 836–847, Aug. 2013.

[4] Y. Yin *et al.*, "Spectral and spatial 2D fragmentation-aware routing and spectrum assignment algorithms in elastic optical networks," *J. Opt. Commun. Netw.*, vol. 5, pp. A100–A106, Oct. 2013.

[5] M. Zhang *et al.*, "Bandwidth defragmentation in dynamic elastic optical networks with minimum traffic disruptions," in *Proc. of ICC 2013*, pp. 3894–3898, Jun. 2013.

[6] L. Gong and Z. Zhu, "Virtual optical network embedding (VONE) over elastic optical networks," *J. Lightw. Technol.*, vol. 32, pp. 450–460, Feb. 2014.

[7] J. Liu *et al.*, "On dynamic service function chain deployment and readjustment," *IEEE Trans. Netw. Serv. Manag.*, vol. 14, pp. 543–553, Sept. 2017.

[8] L. Gong, Y. Wen, Z. Zhu, and T. Lee, "Toward profit-seeking virtual network embedding algorithm via global resource capacity," in *Proc. of INFOCOM 2014*, pp. 1–9, Apr. 2014.

[9] Q. Sun, P. Lu, W. Lu, and Z. Zhu, "Forecast-assisted NFV service chain deployment based on affiliation-aware vNF placement," in *Proc. of GLOBECOM 2016*, pp. 1–6, Dec. 2016.

[10] A. Morton, "Active and passive metrics and methods (with hybrid types in-between)," *RFC 7799*, May 2016. [Online]. Available: https://tools.ietf.org/html/rfc7799.

[11] P. Bosshart *et al.*, "P4: Programming protocol-independent packet processors," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, Jul. 2014.

[12] S. Li *et al.*, "Protocol oblivious forwarding (POF): Software-defined networking with enhanced programmability," *IEEE Netw.*, vol. 31, pp. 12–20, Mar./Apr. 2017.

[13] INT dataplane specification. [Online]. Available: https://github.com/p4lang/p4-applications/blob/master/docs/INT_v2_1.pdf.

[14] Y. Kim, D. Suh, and S. Pack, "Selective in-band network telemetry for overhead reduction," in *Proc. of CloudNet 2018*, pp. 1–3, Oct. 2018.

[15] B. Niu *et al.*, "Visualize your IP-over-optical network in realtime: A P4-based flexible multilayer in-band network telemetry (ML-INT) system," *IEEE Access*, vol. 7, pp. 82 413–82 423, Aug. 2019.

[16] S. Tang *et al.*, "Sel-INT: A runtime-programmable selective in-band network telemetry system," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, pp. 708–721, Jun. 2020.

[17] B. Basat *et al.*, "PINT: Probabilistic in-band network telemetry," in *Proc. of ACM SIGCOMM 2020*, pp. 662–680, Aug. 2020.

[18] S. Tang *et al.*, "Self-adaptive network monitoring in IP-over-EONs: When multilayer telemetry is flexible and driven by data analytics," in *Proc. of OFC 2021*, pp. 1–3, Jun. 2021.

[19] A. Castro *et al.*, "Near-optimal probing planning for in-band network telemetry," *IEEE Commun. Lett.*, vol. 25, pp. 1630–1634, May 2021.

[20] G. Kahe and A. Jahangir, "On the Gaussian characteristics of aggregated short-lived flows on high-bandwidth links," in *Proc. of AINA 2013*, pp. 860–865, Mar. 2013.

[21] D. Kutuzov *et al.*, "Processing of the Gaussian traffic from IoT sources by decentralized routing devices," in *Proc. of SIBCON 2019*, Apr. 2019.

[22] W. Shi, Z. Zhu, M. Zhang, and N. Ansari, "On the effect of bandwidth fragmentation on blocking probability in elastic optical networks," *IEEE Trans. Commun.*, vol. 61, pp. 2970–2978, Jul. 2013.

[23] K. Rezaul and V. Grout, "Identifying long-range dependent network traffic through autocorrelation functions," in *Proc. of LCN 2007*, pp. 692–697, Oct. 2007.

[24] X. Chen *et al.*, "DeepRMSA: A deep reinforcement learning framework for routing, modulation and spectrum assignment in elastic optical networks," *J. Lightw. Technol.*, vol. 37, pp. 4155–4163, Aug. 2019.

[25] R. Stratonovich, *Topics in the Theory of Random Noise*. Gordon and Breach. New York, 1963.

[26] S. Liu and Z. Zhu, "Generating data sets to emulate dynamic traffic in a backbone IP over optical network," *Tech. Rep.*, 2019. [Online]. Available: https://github.com/lsq93325/Traffic-creation/blob/master/README.md?tdsourcetag=s_pctim_aiomsg.