PreTE: Traffic Engineering with Predictive Failures

Congcong Miao[1] Zhizhen Zhong[2] Yiren Zhao[1] Arpit Gupta[3] Ying Zhang[4]
Sirui Li[1] Zekun He[1] Xianneng Zou[1] Jilong Wang[5]

[1]Tencent [2]Massachusetts Institute of Technology [3]UC Santa Barbara [4]Meta [5]Tsinghua University

ABSTRACT

Fiber links in wide-area networks (WANs) are exposed to complicated environments and hence are vulnerable to failures like fiber cuts. The conventional approach of using static probabilistic failures falls short in fiber-cut scenarios because these fiber cuts are rare but disruptive, making it difficult for network operators to balance network utilization and availability in WAN traffic engineering. Our large-scale measurements of per-second optical-layer data reveal that the fiber's failure probability increases by several orders of magnitude when experiencing a rare and ephemeral *degradation* state. Therefore, we present a novel traffic engineering (TE) system called PreTE to factor in the dynamic fiber cut probabilities directly into TE systems. At the core of the PreTE system, fiber degradation facilitates failure predictions and traffic tunnels to be proactively updated, followed by traffic allocation optimizations among updated tunnels. We evaluate PreTE using a production-level WAN testbed and large-scale simulations. The testbed evaluation quantifies PreTE's runtime to demonstrate the feasibility to implement in large-scale WANs. Our large-scale simulation results show that PreTE can support up to 2× more demand at the same level of availability as compared to existing TE schemes.

CCS CONCEPTS

• **Networks** → **Physical links**; *Network simulations*; **Network experimentation**; *Network measurement*; • **Computer systems organization** → **Availability**; • **Computing methodologies** → **Machine learning algorithms**;

KEYWORDS

Wide-area networks, Traffic engineering, Optical Failures, Machine learning, Network optimization

ACM Reference Format:
Congcong Miao, Zhizhen Zhong, Yiren Zhao, Arpit Gupta, Ying Zhang, Sirui Li, Zekun He, Xianneng Zou, Jilong Wang. 2025. PreTE: Traffic Engineering with Predictive Failures. In *ACM SIGCOMM 2025 Conference (SIGCOMM '25), September 8–11, 2025, Coimbra, Portugal.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3718958.3750508

1 INTRODUCTION

In a large-scale Wide-area Network (WAN), fiber cuts are rare and can be caused by natural disasters, human errors, animal bites, or hardware malfunctions. These cuts can take a significant amount of time to repair, especially for submarine fibers, as they require on-site human intervention. Fiber cuts can be disruptive, often resulting in the loss of Terabits of capacity and affecting a large portion of flows. To mitigate the impact of fiber failures, a common practice is to proactively deploy excess network capacity, allowing traffic to be redirected to backup paths when fiber cuts occur [3, 33]. This approach prioritizes availability over utilization, providing high availability during failures at the cost of lower utilization during normal operations [6, 26].

Traffic engineering (TE) is a critical component to guarantee network availability in face of fiber cuts by utilizing the abundant capacity through flow rerouting [2, 6, 10, 17, 18, 21, 24, 26, 29, 31, 36, 39, 41]. It typically involves allocating traffic flows to a set of tunnels in the network to optimize for specific goals[8, 37]. By applying TE with different objectives, one can navigate the tradeoffs between availability and utilization. Recent TE solutions can be divided into two broad categories based on how they handle failures: *reactive* and *proactive*. The reactive solutions, such as NCFlow [2], Flexile [21], update the TE policies in reaction to failures. Specifically, the centralized controller computes a new TE policy after monitoring a link failure and pushes the related changes to affected routers. While these centralized decisions compute effective TE policies, ensuring *high utilization* of available network capacity, they are slow to react to link failures, causing packet loss during convergence even if the network utilization is quite low. To address this issue, there are proposals for proactive TE, such as FFC [26], TeaVar [6], ARROW [41], plan for future uncertainties by leaving a portion of network capacity unused. On detecting failures, they make local decisions to leverage this otherwise unused network capacity to ensure *high availability*. The network operators in cloud WAN prefer proactive approaches in production as ensuring network availability is given higher priority over resource utilization [26].

The proactive approaches plan for future fiber cuts by using static probabilistic failures for each link. Specifically, a recent proactive TE solution, TeaVar [6] assumes static failure probability for each link, i.e., the failure probability of the link is constant across time epochs, and then uses a probabilistic failure model to compute TE policies. Note that the link failure probability is a key factor for proactive TE, and the efficacy of such an approach relies heavily on how accurately they estimate the failure probability. On the one hand, overestimation will result in a conservative TE that underutilizes network capacity to achieve the desired availability. On the other hand, underestimation will cause a more aggressive TE that makes effective use of network capacity but struggles to handle failures, severely affecting availability.

Our measurements of per-second optical-layer data from a large-scale production network challenges the common belief that failure probability is constant over time. Instead, we observe that the failure probability for each link is not stable across time epochs. Specifically, we observe that about 25% of fiber cuts is not abrupt and will

SIGCOMM '25, September 8–11, 2025, Coimbra, Portugal

ACM ISBN 979-8-4007-1524-2/2025/09...$15.00
https://doi.org/10.1145/3718958.3750508

transition to a *degradation* state before experiencing failure. When we observe a degradation signal, the fiber's failure probability will increase to 40% on average, which is several orders of magnitude higher than the common belief [1]. This phenomenon indicates a dynamic failure probability across time epochs, depending on the appearance of degradation signals. As a result, existing TE solutions overestimate the failure probability for epochs with no failures and underestimate it otherwise. Consequently, the TE policies are less effective in making the best use of available network capacity.

In this paper, we present a novel **Pre**diction-based **TE** solution (PreTE), which makes use of failure prediction to achieve the best of both worlds: high utilization and high availability simultaneously. PreTE is a hybrid TE solution which updates traffic tunnels when a degradation signal occurs (*reactive*), and then uses this signal to obtain more accurate failure probabilities and make effective TE decisions to accommodate possible future link failures (*proactive*). PreTE grapples with the algorithmic challenges of making the efficient prediction of future failure probability and also with system-level challenges, such as how to effectively incorporate preparations for future failures into the existing TE framework.

To make the efficient prediction of link failure probability at each time epoch, we first conduct a comprehensive study on our production network to understand the predictability of fiber cuts, and statistically examine and confirm the correlation between the degradation signal and fiber cut via a hypothesis test. We then identify key features of fiber degradation that are strongly correlated with fiber cuts: time, degree, gradient, and fluctuation. We propose using neural network models to capture complex relationships among these features and make an efficient prediction of link failures. For these time epochs without signals, we proof lower failure probability which allows for more aggressive use of paths. In this way, link failure probability is accurately calibrated at each time epoch which allows for more efficient TE policies to proactively accommodate possible future link failures.

The ability to accurately predict failures can also be leveraged to enable a reactive response to preparation for future failures upon detecting degradation signals. Specifically, the TE controller can establish new tunnels for flows suffering high failure probability. It's important to note that establishing tunnels in a large-scale network takes time, which is often a significant contributor to traffic loss during TE convergence after a failure. By setting up tunnels ahead of time, flows can be quickly switched to them after or even slightly before the actual failure occurs, thereby maximizing overall flow availability. Our evaluation shows that PreTE offers more than 2× more demand for a 99% availability threshold compared to existing approaches.

Contributions. Our contributions are as follows:

- We share a comprehensive characterization of real-world fiber cut events including its occurrence and impact, and demonstrate the power of fiber prediction in enabling effective TE decision-making (§ 2).
- We evidence the predictability of fiber cuts via hypothesis test and develop a prediction model using key factors to better estimate failure probability (§ 3, § 4.1).
- We introduce a new reactive component that creates new tunnels when observing degradation signals to effectively accommodate future link failures (§ 4.2,§ 4.3).
- We evaluate the deployability of PreTE on a production network and compare its performance with other systems through extensive testing (§ 5, § 6, § 7).

Although part of the fiber cuts are predictable, PreTE with acceptable prediction accuracy presents a promising results. But there is still a gap with the optimal. Exploring more predictable fiber cuts by introducing additional indicators and designing more efficient models will be our future work. Moreover, we open-source our codebase to facilitate more discussions in the community.

2 BACKGROUND AND MOTIVATION

Today's WAN TE system contains two steps: tunnel routing and traffic split ratio allocation [24]. A tunnel is an end-to-end path between a source-destination pair (i.e., a flow) that traverses one or more physical links. A TE policy can use multiple tunnels to carry the traffic between the two endpoints. Establishing a tunnel requires updating routing configurations for all the intermediate forwarding elements (e.g., routers), and the process can take several minutes for the entire network [26]. For each time epoch (usually five minutes in production environments [17, 20]), a TE policy determines how to map flows to the established tunnels (tunnel selection) and how to spread the traffic for each flow across different tunnels (rate adaptation).

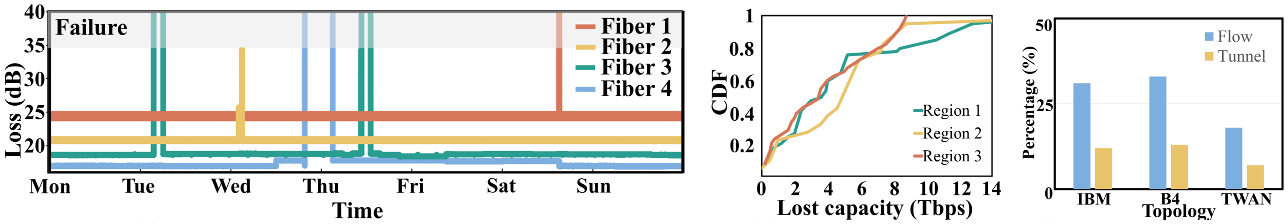
2.1 Curse of Sparsity for Fiber Cuts in TE

In an ideal world with no link failures, computing policies that make the best use of available network capacity are trivial. However, in practice, link failures (i.e., fiber cuts) are inevitable, severely affecting network availability.

Fiber cuts are rare but disruptive events in WANs. Similar to OpTel [28], we collect the *Tx power* and *Rx power* metrics at both ends of the fiber from the optical layer in our production WAN (TWAN) and calculate the transmission loss of fibers. Figure 1(a) illustrates the transmission loss of four fibers that encounter cuts in a typical week. Here, fiber cuts represent the physical line failures within the optical fiber cable, causing the optical signal to be unable to transmit or resulting in a transmission loss increase of 10 dB, which is considerably higher as compared to the healthy state [28]. We observe that fiber cuts are quite rare, with at most two failures for a week. The median number of fifteen-minute epochs [6] for which a link observed any link failure was less than 1%. Figure 1(b) shows the CDF of lost capacity on IP links caused by fiber cuts from three regions. More than 50% of fiber cut events result in the loss of 4 Tbps of IP layer capacity, and the maximum is up to 12 Tbps. This massive loss of capacity affects a lot of flows. Figure 1(c) shows the average number of affected flows and tunnels for a single fiber cut on three typical topologies, i.e., B4, IBM, and our TWAN. The detailed experiment setup is shown in § 6.1. We observe, on B4 topology, 33% of flows and 13% of tunnels are affected when a fiber cut event happens. This massive loss of capacity and a large portion of affected flows bring challenges to high guarantee network availability. Therefore, most existing TE solutions consider link failures in their decision-making. We can broadly divide them into categories: reactive or proactive, depending on how they treat link failures.

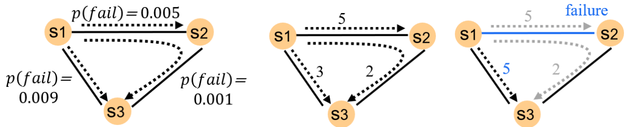
Reactive TE cannot guarantee high availability. Some TE solutions [21, 24, 39] embrace a reactive approach to handle link failures. Specifically, they incorporate a centralized controller to react to

[1] The existing approaches, such as TeaVar [6], always assume a quite low link failure probability with less than 1%.



(a) Transmission loss of four fibers for a typical week (b) Lost IP layer capacity (c) Affected flows and tunnels

Figure 1: Characteristics of fiber cuts: fiber cut failures are rare and disruptive.



(a) Network Topology (b) Throughput=10 (c) Throughput=5

Figure 2: (a) A network of three links, each with 10 units link capacity; (b) A possible flow allocation under conventional TE scheme, i.e., TeaVar [6] with total admissible traffic (i.e., throughput) of 10 units; (c) The redistribution of traffic by TeaVar when link $s1s2$ fails.

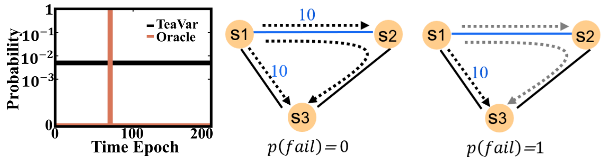
failures by establishing new tunnels (possible) and redirecting the affected traffic to these new as well as existing tunnels to satisfy the bandwidth requirements. However, reaction time is fundamentally limited by factors such as network latencies, establishing new tunnels, computing new TE policies, and noise inherent in production environments. They fail to satisfy bandwidth requirements for a subset of flows for minutes during convergence, affecting their ability to satisfy the availability requirements. Furthermore, packet loss still exists for a long time even if the network utilization is quite low.

Proactive TE suffers from low utilization. Many recent TE solutions [6, 26, 41] embrace a proactive approach to handle link failures. These solutions compute TE policies that guarantee protection against future link failures. Specifically, they proactively leave residual capacity for each tunnel, enabling them to handle more traffic during failure. When a link fails, they only require the switches at the affected endpoints to update the rate adaptation policy, i.e., update what fraction of traffic for the flow gets mapped to which tunnel. The existing tunnels can simply use the residual capacity to mitigate the impact of failures — offering better availability at the cost of link utilization. Since updating rate adaptation policy only requires updating match-action entries at few switches, it is relatively fast to address link failures.

2.2 The Power of Predictive Fiber Cuts

Network operators prefer proactive solutions in production settings as they prioritizes availability over utilization [26]. The recent proactive approaches plan for future fiber cuts by using a probabilistic failure model to compute TE policies. Specifically, a typical proactive TE solution, TeaVar [6] assumes static failure probability for each link, i.e., the failure probability of the link is constant across time epochs. We now explore if it is possible to outperform the existing TE solution. To this end, we explore how TeaVar computes its TE policies and demonstrate how being able to better predict future uncertainties can help improve its performance.

TeaVar as an illustrative example. TeaVar aims at maximizing bandwidth allocation with respect to an availability bound β (i.e., 99%). To accomplish this, TeaVar optimization sets a cap on the maximum bandwidth b_i each flow i can utilize and generates routing rules, such that the network can withstand failure scenarios where the total probability of these scenarios is larger than β. We illustrate TeaVar in Figure 2. Consider a network of three links $s1s2$, $s1s3$, and $s2s3$, each with 10 units capacity and a failure probability of 0.005, 0.009, and 0.001, respectively. There are flows that flow $s1s2$ transfer traffic through one tunnel and flow $s1s3$ transfer traffic through two tunnels. Suppose that the objective of availability bound β is 99%, i.e., no flow sees the loss 99% of the time. Rate limiting the traffic of flows $s1s2$ and $s1s3$ to send at 10 units. Figure 2(b) presents an optimal solution under TeaVar with total admissible traffic of 10 units. We provide a possible flow allocation of flows $s1s2$ and $s1s3$, where the numbers represent the traffic volume each tunnel carries. Both flow $s1s2$ and flow $s1s3$ see no loss when link $s1s2$ is up and at least one of link $s1s3$ and $s2s3$ is up [2]. Any increase of traffic on two flows will break the availability bound. For example, if we slightly increase 0.1 units traffic on flow $s1s2$, there will be a loss of traffic when at least one of link $s1s3$ or $s1s2$ is down with a probability approximate to 1.4%. The probability of failure scenarios that two flows see no loss is at most 98.6%, breaking the availability bound of 99%. Therefore, this solution guarantees the admissible traffic of 10 units by meeting the availability bound of 99%.

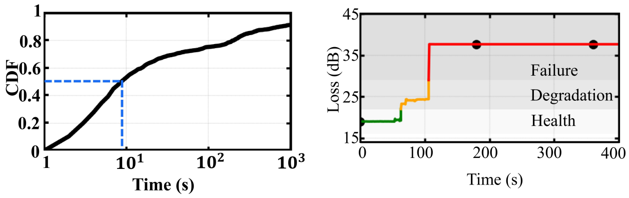


(a) Failure probability (b) Throughput=20 (c) Throughput=10

Figure 3: An oracular TE system with future knowledge of link $s1s2$ and its impact on the total throughput.

Oracular TE system with optimal solutions. We then study how a TE system behaves if it has oracular future knowledge of the link failure, namely the oracular TE system. Here, oracular means that we have known the actual fiber state, i.e., the failure probability is 1 if the link fails, and 0 otherwise. To make the effective comparison to TeaVar, we assume that the oracular TE system only has future knowledge of link $s1s2$ (see Figure 3(a)), i.e., the system can know if and when link $s1s2$ will fail, and the rest settings remain the same as TeaVar. Figure 3(b) presents the optimal solution under the oracular TE system where the link $s1s2$ will not fail in the next time epoch. Here, numbers represent traffic volume each tunnel carries. We observe that the oracular TE system makes full use of link $s1s2$ capacity, indicating more aggressive TE decisions. TeaVar makes

[2] The total probability of link $s1s2$ being up and at least one of link $s1s3$ and $s2s3$ being up, is $(1 - 0.005)(1 - 0.001 * 0.009) = 0.9949$.



(a) Length distribution (b) A typical scenario

Figure 4: Characteristics of fiber degradation. (a) Length distribution of fiber degradation; (b) A typical scenario that a link transitions to a degraded state before experiencing failure.

a conservative traffic allocation that such an allocation keeps link s1s2 underutilized. In contrast, the oracular TE system increases the throughput for both flows $s1s3$ and $s1s2$, and the total throughput is 20 units.

On the other hand, if the link $s1s2$ fails, TeaVar will lose half of the throughput (see Figure 2(c)). Note that the traffic of flow $s1s3$ on tunnel $s1s2s3$ will be rescaled to tunnel $s1s3$. As the oracular TE system has known the future failure, it will make more effective TE decisions to ensure the bandwidth for flow $s1s2$ (see Figure 3(c)). The total throughput is still 10 units. Furthermore, if we know the oracular future knowledge of all links, the TE decisions are more effective to satisfy bandwidth and availability requirements.

Takeaway. This discussion highlights that the effectiveness of such an approach significantly relies on the precision of failure probability estimation. An overestimation will lead to a conservative TE approach that underutilizes network capacity to achieve desired availability, whereas an underestimation will result in a more aggressive TE but struggle to handle failures, severely affecting availability. It is clear that if a TE solution can better predict if and when a link will fail, it can satisfy more intense bandwidth and availability requirements than existing solutions. This observation motivates us to exploit this power of predictability and explore if and how we can improve the predictability of future failure events.

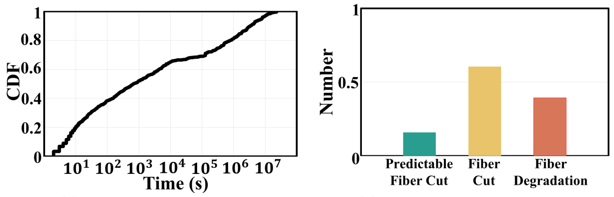
3 EVIDENCING FIBER CUTS PREDICTABILITY

In this section, we firstly demonstrate that fiber cuts are predictable based on the degradation signal (§ 3.1), then characterize critical features that can make a prediction of fiber cuts (§ 3.2), and finally illustrate the performance gain of updating new tunnels during degradation and before cuts (§ 3.3).

3.1 Correlating Degradation with Cuts

Similar to OpTel [28], the fiber degradation is regarded as an increase in the signal transmission loss within the fiber by 3∼10 dB compared to its healthy (normal) state. This observably affects the SNR in the physical layer, but the signal still supports to merit error-free decoding at the destination. The fiber degradation has been little studied, let alone taken for correlating with fiber cuts.

Observing fiber degradation with fine-grained data. Similar to OpTel [28], we have also deployed an optical telemetry system at TWAN to collect one-second granularity optical layer data for about one year. Collecting optical data at such finer granularity may introduce additional noise, such as the loss of data. We apply interpolation methods to complete the missing data. Since some fibers may degrade together because of a common conduit or their geographical proximity. In our work, we consider these fibers as a single entity and regard them as a single scenario. Our empirical observation in Figure 4(a) shows that fiber degradations are always ephemeral with 50% of them lasting for less than 10 seconds. This observation validates the effectiveness of our system in capturing degradation events. Such fine-grained telemetry data enables us to monitor short-lived transient activities in optical links that precede link failures. We zoom in on a typical failure event of fiber1 in Figure 1(a). As shown in Figure 4(b), the failure event is not abrupt for this link. Instead, we see that this link transitions to a degraded state before experiencing failure. This phenomenon can occur for a variety of reasons, including accidental fiber conduit breaking by bulldozers, fiber being crewed by animals, and unintentional damage by network operators. However, traditional systems [13, 35, 40] collect optical data at the minute level. For example, a recent study shows the collection of optical transmission loss data at a middle frequency (i.e., several minutes) [40]. Here, we use a 3-minute data granularity as an example (see black circle in Figure 4(b)). We find that collecting data at this granularity makes it hard to capture such a short-lived transient activity. The fine-grained telemetry data motivates us to dig deeper to study the relationship between degradation events and fiber cuts.



(a) Length distribution (b) Number of fiber events

Figure 5: Relationships of fiber degradations and fiber cuts.

Fiber degradation and fiber cuts relationship confirmation with statistical hypothesis test. Prior work has presented the hypothesis by monitoring changes in optical signal quality [13]. However, there are no formal verification to prove this hypothesis. Figure 5(a) shows the time length distribution of fiber cuts happening after fiber degradations. We observe that 60% of fiber cuts happen after degradation signals within 10^3 seconds and 20% are more than several days. We believe that these cuts happening several days after degradation will not be relevant to its previous degradation. Based on feedback from network operators, for these fiber cuts closely related to degradations, the time intervals should be within several minutes. Here, we use 5 minutes to accommodate a TE period [6, 26] and define them as *predictable fiber cuts*. We then study the number of each type of fiber event (normalized), as shown in Figure 5(b). About 25% (# of predictable cuts/# of fiber cuts) of fiber cuts are predictable using degradation signals, showing a close relationship between them. To confirm this observation, we apply a statistical hypothesis test on the relations between them and set the null hypothesis to be: *fiber cuts are not related to fiber degradations.* We apply a chi-square test. The results show a p-value with less than 10^{-50}. Considering a p-value threshold of 0.01, the null hypothesis is rejected, indicating fiber cuts and degradations are related in a statistically significant manner. We also presented the statistics that the null hypothesis cannot be rejected (see Appendix A.1).

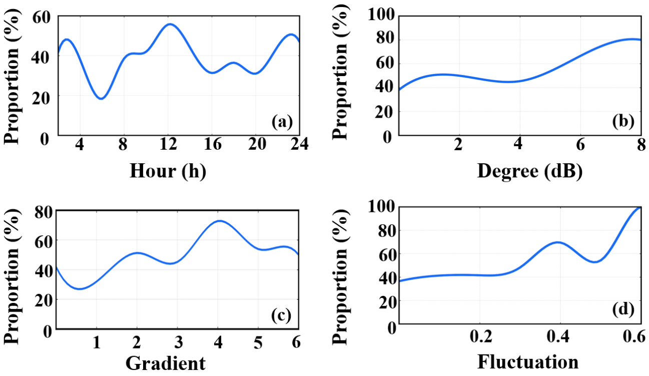


Figure 6: The failure proportion among four critical features, i.e., time, degree, gradient, and fluctuation.

3.2 Characterizing Critical Features

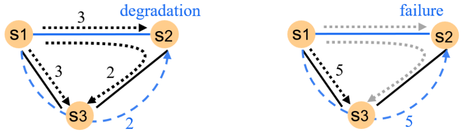
Our measurements have shown that despite fiber cuts and degradation signals being related in a statistically significant manner, only 40% of fiber degradation will lead to fiber cuts. A naive approach to estimate the failure probability is to simply assume a fixed failure probability when fiber degradation occurs. However, as a key input in probabilistic TE, an accurate estimation of failure probabilities will lead to more effective TE decisions. Since there is valuable time series information (e.g., timestamp, degradation magnitude) of degradation events, we argue a fixed failure probability is not optimal. For example, our measurement results show that degradation events with frequent fluctuations are more likely to evolve into failures. This motivates us to characterize critical features of fiber degradation.

Observing critical features. Taking critical features of fiber degradations as inputs stimulates a good prediction system. In addition to the intrinsic features (i.e., region, length) of a fiber that have been used in previous work [38], we learn features from a series of transmission losses during the degraded state to study if these features are more likely to make good discrimination of degradations, i.e., which types of degradations are more likely to fail. Below are the features we have studied, and Figure 6 shows the failure proportion at the specific value of these features. Here, the failure proportion refers to the number of fiber cuts to fiber degradations at a specific x-axis value.

- *Time*: The happening time shows different failure possibilities. The failure probability is about 60% when we observe a fiber degradation event at 12 am, but it degrades to 20% at 6 am. We guess the high proportion is due to unplanned human intervention.
- *Degree*: The degree indicates changed loss when a fiber transits from a healthy state to a degraded state. We observe that a larger degree always leads to a higher failure probability.
- *Gradient*: The gradient indicates the loss changes between the adjacent values during a degraded state. We observe that a small gradient of a degradation event is less likely to evolve into failures. This is expected since these degradation events may be caused by the long-term aging of the optical fiber.
- *Fluctuation*: The fluctuation indicates the frequency of loss changes during degradation. We only consider the fluctuations larger than 0.01 dB between the adjacent values to filter out the noises. We observe a higher failure probability when there are frequent fluctuations during the fiber degradation.

Characteristic	Gradient	Time	Degree	Fluctuation
P-value	1.1×10^{-7}	10^{-6}	2.2×10^{-13}	10^{-11}

Table 1: Statistical hypothesis test on critical features.



(a) Degradation reaction (b) Failure reaction

Figure 7: Preparing for failures by establishing new tunnels.

Feature confirmation with statistical hypothesis test. The results in Figure 6 show that these features affect the failure probability. To confirm this observation, we apply a statistical hypothesis test on these features. Taking the gradient feature as an example, we set the null hypothesis to be: *fiber cuts are not related to gradient values*. As the gradient value is continuous, we perform equal-width binning, which divides the range of values into intervals with equal width, and calculates the number of values that fall into each interval. We then apply a chi-square test. Table 1 shows a p-value of 1.1×10^{-7}. Considering a p-value threshold of 0.01, the null hypothesis is rejected, indicating the fiber cuts and gradient values are related in a statistically significant manner. We observe a similar trend for other features with p-values much less than 0.01. Our hypothesis test results confirm the results in Figure 6.

3.3 Prepare for future failure events

As we can know part of future failure events, we argue that we can make more effective TE decisions by incorporating tunnel selections, i.e., creating new tunnels for the flows that suffer from fiber degradations. We assume the settings are the same as TeaVar in Figure 2(a), except for the state of link $s1s2$. We assume link $s1s2$ suffers from fiber degradation, indicating a high failure probability in the future. By observing this, we create new tunnels for both flow $s1s2$ and $s1s3$ to prepare for future failures, i.e., flow $s1s2$ has new tunnel $s1s3s2$ and flow $s1s3$ remains the same because there is no new path. Then the centralized controller calculates new TE policies and distributes them to switches. The numbers represent the traffic volume each tunnel carries (see Figure 7(a)). We observe that the newly created tunnel absorbs the traffic of flow $s1s2$. Since TeaVar only supports 5 units traffic by rate adaption as link $s1s2$ fails in Figure 2(c), Figure 7(b) shows our approach still supports 10 units by rerouting traffic to newly created tunnels. Therefore, our approach maintains high network throughput by leveraging degradation signals.

3.4 Challenges

Although we have the ability to know part of the future failure events, there are two fundamental challenges that one needs to realize the ideas described above.

Challenge1: ephemeral optical degradations make it difficult to achieve accurate predictions of failures. As 40% of fiber degradations lead to the final fiber cuts, we should make the best of the degradation information to conduct prediction. However, our measurements using the one-second optical telemetry system show that optical degradation events are rare and ephemeral. A small amount of degradation dataset makes it hard to characterize critical features of the degradations and model the complex relationships between degradations and failures. Furthermore, the characterized features are always heterogeneous. It is challenging to effectively

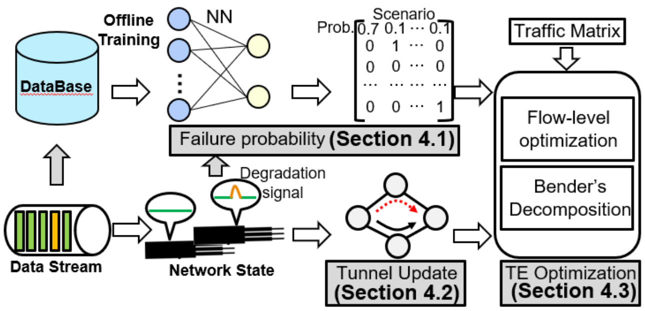


Figure 8: PreTE's high-level design.

aggregate these features to make an accurate prediction of future failures.

Challenge2: incorporating preparation for future failures offers new system-level problems. Since the presence of fiber degradation suggests a higher likelihood of failure, it is important to proactively prepare for future failures to reduce their impact. A promising approach is to establish new tunnels to absorb the network traffic. However, it is very challenging to determine a specific fraction of the flows, update tunnels for these flows on healthy fibers, and then seamlessly incorporate these tunnels in the TE optimization framework to minimize traffic loss.

4 PRETE SYSTEM DESIGN

Figure 8 presents PreTE's high-level design, which emphasizes on two critical aspects in today's TE system, i.e., probabilistic failure model for TE calculation and tunnel routing. PreTE develops an NN model to better estimate failure probability (§ 4.1). As fiber degradation occurs, it will trigger the reactive process that PreTE makes preparation (e.g., creating new tunnels) for future failures (§ 4.2). PreTE seamlessly incorporates the newly established tunnels into a proactive TE optimization framework to address failures whose generated TE policies guarantee protection against faults (§ 4.3).

4.1 Calibrate link's failure probability

As a critical input of TE optimization, the efficacy of existing TE schemes relies heavily on how accurately they estimate failure probability. Previous work [6] showed a fixed estimated failure probability represented as $p_i = \frac{n}{T}, \forall i \in [1, T]$ and $p_i \ll 1$ where n is the number of failures and T is measured time epochs. However, our measurement results challenge this basic assumption and we show that failure is more likely to happen when a degradation occurs (much higher than p_i). Thus, we break the failure probability of the fiber in next TE period into two parts, conditionally depending on if there is a fiber degradation.

4.1.1 Failure Probability When Degradation Occurs.

Given that 40% of fiber degradations result in failures, a naive approach to estimating failure probability after degradations is to simply assume the failures follow a geometric distribution, i.e., the failure probability is fixed and consistent after each fiber degradation. However, our statistical hypothesis test in § 3.2 has indicated that there are several critical features which are closely related to the failure probability. It is possible to discriminate between different degradation events to better estimate failure probability. This motivates us to make more accurate failure probability predictions based on these features. Therefore, we establish a prediction system to model the fiber degradation event. The problem can be formalized as follows: *Given a set of features of a fiber degradation event, the model outputs the failure probability of the next TE period.*

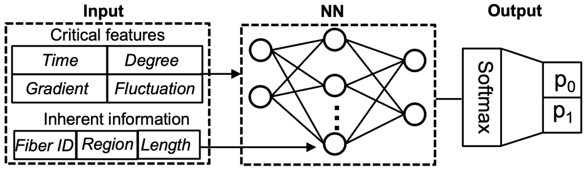


Figure 9: Neural network algorithm.

Neural network modeling. Upon receiving critical features of the fiber degradation event and intrinsic features (i.e., region, length) of the fiber, it is challenging to aggregate these variables since they are heterogeneous and their relationships are complex. Traditional models such as decision tree (DT) are not effective in modeling such complex relationships. To overcome this, as illustrated in Figure 9, PreTE uses a neural network (NN) [16] to model relationships with a manageable number of adjustable parameters, θ, which we refer to as NN parameters. Neural networks have shown great power in capturing complex relationships among inputs, as they do not rely on hand-crafted features and instead use "raw" observation values to model these relationships [25, 27]. Directly training an NN algorithm for each fiber (i.e., one-model-one-fiber approach) is impractical with low data samples of fiber degradation generated by each fiber. We introduce a multi-layer perception (MLP) approach that inputs critical features of the fiber degradation in the first layer and inherent information in the second layer. This approach always performs better since 1) there is sufficient data to train the NN model; 2) it not only learns from the previous samples of current fiber but also the information from other fibers. We refer the reader to Appendix A.2 for implementation details. Throughout the NN training process, as there exists a slight imbalance between the two classes (4:6), we adopt the oversampling approach to address the imbalance and achieve equilibrium between the two classes. The output of NN is the normalized probability distribution $\vec{p}$, i.e., $p_0 + p_1 = 1$ where p_0 and p_1 represent the estimated probability of normal and failure, respectively. We use the predicted label $\hat{y} = argmax_p(p)$ to evaluate the performance of NN algorithm in § 6.3.

4.1.2 Failure Probability without Degradations.

We divide T link up-time epochs into the time epochs with and without the fiber degradation respectively and make a simplified assumption that the degradation probability p_d is fixed and consistent across time epochs. Note that the fiber degradation is smaller than fiber cuts and rare, i.e., $p_d \ll 1$ (see Figure 5(b)). The portion of fiber cuts preceded by a degradation event (i.e., predictable fiber cuts) is α, and the portion of unpredictable fiber cuts is $1 - \alpha$.

Theorem 4.1 (Failure Probability without degradation). *Given parameters (T,p_i, p_d, α) and assuming that unpredictable fiber cuts follow geometric distributions, the failure probability is lower (i.e., $(1-\alpha)p_i$) when we do not observe any fiber degradation.*

Please see Appendix A.3 for proof.

We observe that the failure probability is lower and closely related to α when degradation does not occur. As α becomes larger showing more predictable fiber cuts, the failure probability is lower. If α equals 0 (i.e., all fiber cuts are not predictable), the failure probability is p_i and PreTE degrades to the existing work [6]. In contrast, if α equals 1 (i.e., all fiber cuts are predictable), the failure

TE Input	$G(V,E)$	Network graph with routers V and links E.
	$f \in F$	Set of flows F.
	c_e	Bandwidth capacity of link $e \in E$.
	d_f	Bandwidth demand of flow f in a TE interval.
	T_f	Set of pre-established tunnels for flow f.
	$L(t,e)$	1 if tunnel t uses link e and 0 otherwise.
	β	Target availability level for flow.
Additional PreTE Input	$s \in S$	Set of degradation scenarios s.
	$q \in Q_s$	Failure scenario q in degradation scenarios s.
	p_q	Probability of failure scenario $q \in Q_s$.
	$Y_f^s \in Y^s$	Set of newly established tunnels for flow f under degradation scenarios s.
	$T_{f,q}$	Available tunnels for flow f under scenario q.
	$Y_{f,q}^s$	Available tunnels of Y_f^s under scenario q.
Auxiliary variables	$l_{f,q}$	Loss of flow f under scenario q.
	$\delta_{f,q}$	1 if loss of flow f meets the target Φ under scenario q, 0 otherwise.
Output	$a_{f,t}$	Allocated bandwidth on tunnel t for flow f.
	Φ	Maximum loss under β across flows F.

Table 2: PreTE notations.

probability is 0. and PreTE can make more aggressive TE policies. For example, as shown in Figure 3(b) that the failure probability of link $s1s2$ is 0, we can aggressively allocate the traffic on this link to meet its capacity. Generally, the failure probability of PreTE is closer to the oracular knowledge than existing TE schemes. Being able to get the estimated failure probabilities closer to the oracular knowledge, PreTE can make more effective TE policies.

Summary. We obtain the integrated failure probability in each TE period, represented as:

$$p = \begin{cases} p_{NN} & (degradation\ occurs) \\ (1-\alpha)p_i & (otherwise) \end{cases} \tag{1}$$

The failure probability of each fiber is time-variant. When we observe the fiber degradation, we utilize the failure probability output by the NN model. It is important to note that when several fibers experience degradations concurrently, we batch these events from fibers and feed them into the NN model to simultaneously predict failure probabilities. On the other hand, we use the failure probability generated by the geometric distribution.

4.2 Updating TE Tunnels

As each fiber degradation will lead to a high failure probability and each fiber cut will affect a lot of flows as well as tunnels, PreTE takes advantage of the degradation signal to make preparations (e.g., creating new tunnels) to address future failures. The TE notation is listed in Table 2.

Tunnel initialization. We model the WAN as a directed graph $G = (V, E)$, where the vertex set V represents edge routers and edge set E represents IP links. For a source-destination site pair f (or "flow"), it should rely on a set of paths (or "tunnels") T_f to route the flow traffic. We use both k-shortest path routing and fiber-disjoint routing algorithms to establish tunnels over the IP layer topology. We ensure that at least one residual tunnel exists for every flow under each failure scenario.

Reacting to fiber degradation. Since a lot of flows and their tunnels will be affected by future failures, PreTE will create new tunnels for these flows that suffer from the fiber degradation event. For a specific degradation event, the whole procedure of this process is shown in Algorithm 1. We first initialize the newly established tunnel set to $\emptyset$ (lines 3). For a specific fiber degradation event, we then delete the link e which is suffering from the fiber degradation events to obtain a new graph $G'(V, E)$ (line 4). For each flow $f \in F$, we determine whether its tunnels are traversing the degraded link e and get the total number of affected tunnels (line 6). If the flow suffers a high failure probability, new tunnels Y_f^s are established from the new graph $G'(V, E)$ to route the traffic. In this way, the path of newly established tunnels will be disjoint with the degraded fiber (lines 7-8), and these tunnels are added to the set Y^s (line 9). Once the failure is repaired or the failure does not happen after a TE period of degradation, the tunnel is then updated to its original state.

Algorithm 1 TE tunnels update for a degradation event

```
 1: procedure CALCULATE TUNNELS FOR AFFECTED FLOWS
    ▷ Input: $T_f$: pre-established tunnels for flow $f$ in $G(V,E)$
 2:          $e$: Degraded link under degradation scenario $s$
    ▷ Output: $Y^s$ : newly established TE tunnels
 3:     Initialize tunnel set $Y_s \leftarrow \emptyset$;
        ▷ Step 1: Obtaining new graph under degradation scenario s
 4:     $G'(V,E) \leftarrow$ Delete link $e$ which is degraded in WAN graph $G(V,E)$;
 5:     for each $f$ in $F$ do
            ▷ Step 2: Establishing tunnels for flows f suffering degradation
 6:         $\Lambda \leftarrow$ Get tunnel number if tunnels $T_f$ traverse link $e$;
 7:         if $\Lambda > 0$ then
 8:             $Y_f^s \leftarrow$ Establish new $\Lambda$ tunnels for flow $f$ from $G'(V,E)$;
 9:     $Y^s$.add($Y_f^s$)
10:     return $Y^s$
```

4.3 Optimizing Traffic Flow Allocations

This section describes how PreTE formulates a new optimization problem that seamlessly incorporates the preparations for future failures into a unified framework. PreTE builds on the recently proposed Flexile's optimization framework [21].

TE optimization period. Traditionally, the TE optimization is performed at a regular interval (e.g., every 5 minutes [6, 17, 41]) to respond to fluctuations in the traffic demand. However, in PreTE, in addition to the regular optimization schedule, the TE optimization will also be performed in response to the degradation signal. This approach makes our work quite different from the previous work [6, 21, 41].

TE input. For a WAN graph $G = (V, E)$, the capacity of edge e is c_e. A traffic demand d_f exists between a source-destination site pair, which will be routed on a set of pre-established tunnels T_f. We introduce the degradation scenario s to represent whether there is a fiber degradation in this time epoch. Suppose a network with N fibers. The degradation scenario $s = (s_1 = \hat{s}_1, \cdots, s_N = \hat{s}_N)$ is a binary vector where each number $s_n = \{0, 1\}$ and 1 represents the degradation of fiber n. Note that the degradation scenario with all variables to be 0 indicates that no fiber degradation happens. Given the degradation scenario s, the failure probability of fiber n is represented as $p_n = s_n p_{NN} + (1 - s_n)(1 - \alpha)p_i$. The value of p_n is p_{NN} if $s_n = 1$, and $(1 - \alpha)p_i$ if $s_n = 0$. This result is the same as Eqn. 1. There is a set of failure scenarios $q \in Q_s$. Let $p_{\hat{q}}$ be the probability of failure scenario $\hat{q} = (\hat{q}_1, \cdots, \hat{q}_N) \in Q_s$. The probability can be generate by $p_{\hat{q}} = P(q_1 = \hat{q}_1, \cdots, q_N = \hat{q}_N) = \prod_{n=1}^{N} (\hat{q}_n p_n + (1 - \hat{q}_n)(1 - p_n))$ where $\hat{q}_n = \{0, 1\}$ and 1 represents failure event, for each n. Note that the set of failure scenarios Q_s and the failure probability of $p_{\hat{q}}$ change over time, largely depending on whether fiber degradation occurs and which fiber is degraded. Furthermore, for pre-established tunnels T_f and newly established tunnels Y_f^s, $T_{f,q}$ and $Y_{f,q}^s$ are available tunnels for flow f under failure scenario q.

Optimization goal and constraints. For flow f and target availability level β, we define a loss upper bound $\phi_{f,\beta}$ which means there are a set of failure scenarios that together occur with probability β where flow f sees a loss less than $\phi_{f,\beta}$. To achieve the fairness of the whole network, we define a global loss function as the maximum loss across flows: $\Phi = \max_{f \in F} \phi_{f,\beta}$. Recall that our goal is to achieve bandwidth allocation in a way that maximizes the allocated bandwidth that meets the demand of flow. In other words, we should minimize the global loss Φ fraction of demand not satisfied for a specified level of availability β. We present the formulation which is shown below.

$$\min_{\delta,a,l,\Phi} : \quad \Phi \tag{2}$$

$$s.t.$$

$$\forall e : \Big(\sum_{f \in F, t \in T_f} a_{f,t} + \sum_{f \in F, t \in Y_f^s} a_{f,t}\Big) L(t,e) \le c_e \tag{3}$$

$$\forall f, q : \sum_{t \in T_{f,q}} a_{f,t} + \sum_{t \in Y_{f,q}^s} a_{f,t} \ge (1 - l_{f,q}) d_f \tag{4}$$

$$\forall f : \sum_{q \in Q_s} \delta_{f,q} p_q \ge \beta \tag{5}$$

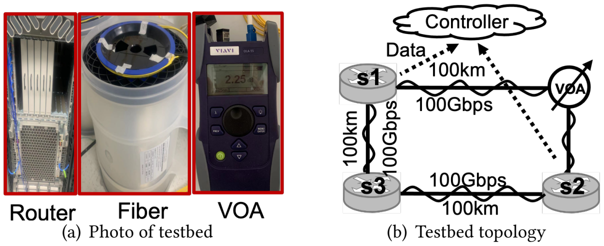
$$\forall f, q : \Phi \ge l_{f,q} - 1 + \delta_{f,q} \tag{6}$$

$$\forall f, q : \delta_{f,q} \in \{0, 1\} \tag{7}$$

$$\forall f, q, t : 0 \le l_{f,q} \le 1, 0 \le a_{f,t} \tag{8}$$

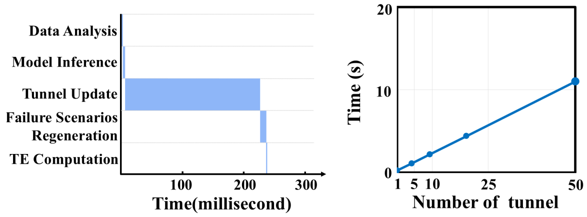
Our objective function (2) is to minimize the maximum loss across all flows. Constraint (3) states that no link should be overloaded. Constraint (4) states that the sum of the bandwidth of available tunnels for flow f under scenario q should be larger than f's demand. Constraint (5) guarantees that the total probability of failure scenarios that are less than Φ meets the target availability level β. Constraint (6) can be treated in a binary way. If $\delta_{f,q} = 1$ meaning loss of flow f meets the target Φ, i.e., $\Phi \ge l_{f,q}$, the constraint (6) holds since it can be simplified to $\Phi \ge l_{f,q}$. Otherwise, if $\delta_{f,q} = 0$ meaning we could take little care of the loss in such failure scenarios, the constraint (6) can be simplified to $\Phi \ge l_{f,q} - 1$. Since the right-hand side of the equation is less than 0 because of constraint (7), i.e., $0 \le l_{f,q} \le 1$, constraint (6) holds under any conditions. Constraint (7) states $\delta_{f,q}$ is binary and 1 represents the scenario q meets the loss bound for flow f, and Constraint (8) states the normalized loss is between 0 and 1, and the allocated bandwidth is non-negative.

A unified framework. We now demonstrate how PreTE flexibly addresses fiber degradations in a unified framework. As shown in Eqn. 3, new tunnels Y_f^s are established with allocated bandwidth $a_{f,t}$ to absorb traffic of flows in advance. Since PreTE adds new tunnels rather than increases network capacity [41], it requires that no link should be overloaded under the current bandwidth capacity after allocating bandwidth for these new tunnels. Therefore, the sum of the bandwidth of pre-established and newly established tunnels for flow f should be lower than the bandwidth capacity of each link. Note, Eqn. 3 can also fit the scenario when we do not observe fiber degradation by setting Y_f^s to $\emptyset$. The Eqn. 3 can be transformed into $\forall e : \sum_{f \in F, t \in T_f} a_{f,t} L(t,e) \le c_e$. As failure happens under scenario q, the traffic is quickly rerouted to the residual (available) tunnels. Note that there are additional newly established tunnels if there is a degradation before the failure. As well, PreTE should guarantee that the sum of the bandwidth of residual pre-established and newly established tunnels for flow f under scenario q should be larger than f's traffic requirement (see Eqn. 4). This equation ensures that PreTE is a proactive approach to address failures. Similarly, Eqn. 4 also fits the scenario when we do not observe fiber degradation by setting $Y_{f,q}^s$ to $\emptyset$. The Eqn. 4 can be transformed into $\forall f, q : \sum_{t \in T_{f,q}} a_{f,t} \ge (1 - l_{f,q}) d_f$.



(a) Photo of testbed (b) Testbed topology

Figure 10: PreTE 's production-level testbed.



(a) Time series (b) Update time

Figure 11: Latency evaluation.

Efficiently solving constraints with Benders. The binary variable $\delta_{f,q}$ in constraints introduces a mixed-integer programming (MIP) formulation which is hard to solve, especially in the large-scale networks. We use Benders' decomposition algorithm [12] to efficiently solve the MIP problem. Please see Appendix A.4 for more details.

5 TESTBED EVALUATION

Testbed setup. We use a production-level testbed to faithfully evaluate PreTE in a real-world setting. Figure 10 shows our testbed set up with three routers, a variable optical attenuator (VOA), and hundreds of kilometers fiber. Here, we only focus on hardware devices closely related to our work. We implement a prototype of network topology (i.e., Figure 2(a)). Figure 10(b) shows its optical layer view where the fiber length is about 100 km, and each wavelength passing through the fiber between a pair of sites provides 100 Gbps capacity for each IP link. We introduce a VOA between $s1$ and $s2$ to allow us to manually adjust the power of the optical signal passing through it. Therefore, we are able to reproduce a typical scenario of a fiber event in the production network (see Figure 4(b)), where a healthy (0-65 seconds) fiber transitions to a degraded state (65-110 seconds) before experiencing failure (110-400 seconds). The controller collects optical data (i.e., *Tx power* and *Rx power*) at both ends of the fiber to monitor the fiber state. All hardware devices and software (e.g., controller) in the testbed are identical to TWAN, demonstrating the ready deployment of PreTE.

Quantifying latency. The controller analyzes the optical data in real time. When there is a fiber degradation, the controller will proceed with a sequence of processes, including model inference, tunnel update, failure scenarios regeneration, and TE computation. We then quantify the latency of each process with the controller deployed in a server with 32-Core CPU processor and 256 GB RAM. Figure 11(a) shows the time series of this experiment, with each rec-

Topology	#Fibers	#IP links	#Tunnels	#Traffic Matrix
IBM	23	85	340	24
B4	19	52	208	24
TWAN	O(50)	O(100)	O(100)	24

Table 3: Network topologies used in our simulations.

tangle representing an operation for some period. The end-to-end latency in our testbed is less than 300 milliseconds. The controller takes a short time to analyze the optical data to detect the fiber degradation. Then the degradation data is fed into the NN model to predict failures. The NN model inference only takes several milliseconds since the time-consuming NN model training is already carried out offline, while the TE computation time is also minimal (see Figure 16). It only takes about ten milliseconds to change the failure scenario when the failure probability of the fiber changes drastically. We observe that the majority of time is spent on the establishment of new tunnels. The update time is calculated between the controller sending configurations to update switches and receiving acknowledgment from switches. This finding motivates us to study the time overhead of creating different numbers of tunnels. Figure 11(b) shows a linear relationship between the update time and tunnel number. Based on the feedback from the engineer, this phenomenon is attributable to their choice to serialize the creation of tunnels, which guarantees a consistent allocation of resource costs at the controller. Despite the serial update of tunnels, this process takes about 5 seconds to update 20 tunnels. This phenomenon shows that we are sufficient to prepare for failures since the cost of total time is lower than the time interval between degradations and fiber cuts in most cases (see Figure 5(a)). Even if there is a large number of tunnels to be updated (e.g., one hundred), it is possible to implement a batch strategy (e.g., update a dozen tunnels at a time) [15] to reduce the overall time required.

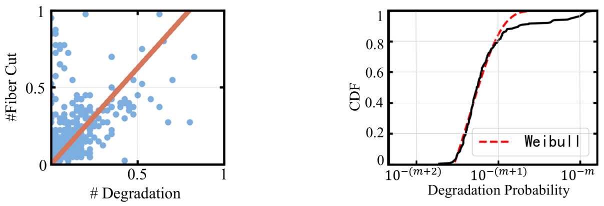
6 LARGE-SCALE SIMULATIONS

We conduct large-scale simulations to demonstrate the performance gains of PreTE. Our simulation framework is implemented with the Julia [5] and Gurobi solver [1]. We show PreTE supports more than 2× more demand at the same level of availability (§ 6.2). By characterizing critical features and developing a new model, we show PreTE achieves high prediction accuracy (§ 6.3). Finally, we present the sensitivity analysis, such as performance gains of creating new tunnels in the degraded state (§ 6.4).

6.1 Setup

Benchmark TE schemes. ECMP [7] serves as a baseline in our evaluations. The existing state-of-the-art TE schemes can be divided into two categories. The proactive approaches include (1) FFC [26], which guarantees no traffic loss under k failures. We evaluate both k=1 and k=2 cases and refer to them as FFC-1 and FFC-2, respectively. (2) TeaVaR [6], a failure-aware probabilistic TE scheme that provides a probabilistic guarantee. (3) ARROW [41], which incorporates optical restoration with eight seconds restoration latency. As for reactive approaches, they are slow in reacting to link failures, affecting availability even if the network utilization is low. We provide a recent work as a representative case: (1) Flexile [21], which uses the flow-level optimization framework.

Topologies and traffic matrix. We evaluate PreTE on three topologies: TWAN [3], B4, and IBM. For TWAN topology, we use a subset of the optical-layer topology in production. For B4 and IBM, we take topologies in [24], which are regarded as optical-layer topologies and use distributions in [41] to generate IP-layer topologies. For each node pair, we generate 4 tunnels using both fiber-disjoint routing and k-shortest path. The details are shown in Table 3.



(a) Number (b) Degradation probability

Figure 12: (a) Normalized number of degradations and failures; (b) CDF of degradation probability.

Quantifying probability of degradation and failure scenarios. For each fiber, we count the number of degradation signals and fiber cuts. Figure 12(a) shows an approximately linear relationship between fiber degradations and fiber cuts. For simplicity, we use a linear function to fit our empirical data. We then study the distribution of degradation probabilities among fibers. Figure 12(b) shows that the degradation probabilities of fibers differ by orders of magnitude. The exact value m is not shown for confidentiality reasons. We use a Weibull distribution to well fit the shape of our measurement data.

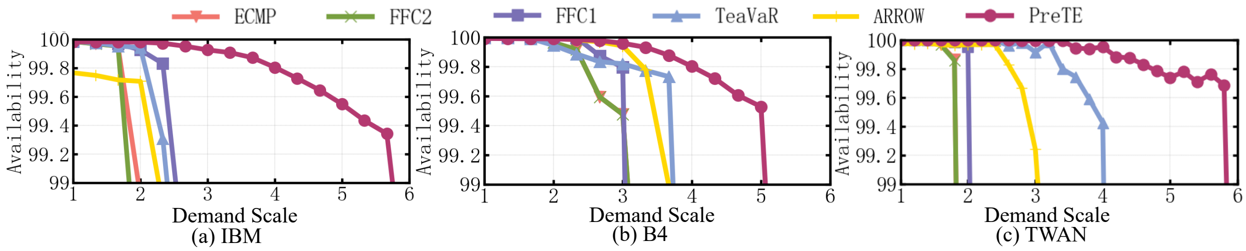
For TWAN, we use the data collected in production. However, for B4 and IBM, generating the degradation and failure probability of a link using a separate Weibull distribution is not reasonable due to their linear relationship. We use a Weibull distribution (shape=0.8, scale=0.002) to generate the degradation probability p_d and then calculate failure probability p_i based on their linear relationship. The scaling property of the Weibull distribution indicates that the failure probability also follows the Weibull distribution, showing consistency with previous work [6]. The obtained failure probability p_i is used in the state-of-the-art TE system. However, PreTE uses Eqn.1 to obtain the failure probability, depending on if fiber degradation occurs (α = 25% from empirical data). By obtaining the degradation and failure probability of each fiber, the probability of degradation and failure scenarios is generated (see § 4.3). We select degradation and failure scenarios based on the specific cutoff values.

6.2 Availability and Satisfied Demand Gains

Availability is a key indicator that cloud providers are concerned about, characterizing SLAs for cloud services. For proactive approaches (i.e., FFC, TeaVar, ARROW, PreTE), we used the method proposed in [6] to obtain the availability. Here, for ARROW, we additionally take the affected flows into consideration during the eight-second restoration. As for reactive approaches (i.e., Flexile), we consider each failure separately to solve the TE formulation and then incorporate the affected flow during the recomputation to obtain the availability.

Impact of demand scaling on availability. Figure 13 summarizes the results by depicting the demand scale and corresponding availability. We focus on the availability region larger than 99% according to service requirements [18]. We observe that PreTE maintains high availability levels as the demand matrix is scaled for all topologies. In particular, we observe that on IBM topology,

[3] The exact value of TWAN is not disclosed for confidentiality reasons.



(a) IBM (b) B4 (c) TWAN

Figure 13: Availability vs. demand scales for PreTE and state-of-the-art TE schemes.

	PreTE's gain in terms of satisfied demand				
	Reactive	Proactive			
Availability	Flexile	ECMP	FFC	TeaVaR	ARROW
99.95%	2.7×	1.6×	1.6×	2.0×	NA
99.9%	3.3×	2.0×	1.7×	1.7×	NA
99.5%	2.1×	3.0×	2.1×	2.5×	2.5×
99%	1.5×	3.4×	2.4×	2.4×	2.8×

Table 4: PreTE's gain at different availability levels [18] on IBM topology. 'NA' means not applicable.

	P	**R**
Teavar	≈0	≈0
Statistic	0.45	0.37
DT	0.68	0.53
NN(Our)	**0.81**	**0.81**

Table 5: Performance.

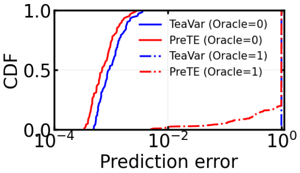


Figure 14: Error.

PreTE can guarantee 99% availability, even if the traffic demand is scaled by 5.7×, while TeaVar and FFC1 only sustain at 2.3× demand at 99% availability. As a result, PreTE provides more than 2× gain in throughput compared to the proactive approaches, e.g., TeaVar, without sacrificing 99% availability. As for higher availability levels, i.e., 99.95%, PreTE still maintains high throughput. However, the availability of ARROW cannot achieve 99.95% even with a demand scale of 1, as ARROW takes eight seconds to complete the end-to-end restoration. A subset of flows that experience failures will suffer persistent packet loss throughout the restoration process, which affects the availability of the flows. We observe a similar trend for both B4 and TWAN topology.

Table 4 shows PreTE's gain with respect to existing proactive and reactive TE approaches at different availability targets [18] on IBM topology. We observe a similar trend for B4 and TWAN topologies in Figure 13. At 99.95% availability, the performance gain of PreTE compared to the reactive approach, i.e., Flexile, is significant at 2.7×. This performance gain is larger than that of proactive approaches such as FFC and TeaVar, due to the prolonged duration required by the reactive approach to address failures. For example, Flexile incorporates a centralized controller to compute new TE policies to address failures. This approach is slow to react to link failures, causing packet loss during convergence even if the network utilization is quite low. PreTE improves the network throughput by 2.0× compared to the proactive approach, i.e., TeaVar under such a high availability level, indicating the effectiveness of our proposed PreTE to achieve the best of both worlds: high utilization and high availability simultaneously. Similarly, at lower availability levels, PreTE still shows a promising performance gain.

6.3 Impact of Prediction Accuracy

Impact of models on prediction accuracy. PreTE characterizes critical features and develops an NN algorithm to better estimate failures. We evaluate the performance of NN model on capturing fiber cuts using real-world data from TWAN. We regard a fail after degradation as positive, negative otherwise. For a binary classification problem, we evaluate the accuracy of PreTE's inference in terms of precision (P) and recall (R)[4]. We compare our NN model with traditional models: (1) TeaVar [6], a naive approach without using the degradation signal to predict failures; (2) Statistic model, which models failures based on the statistical relationship between degradations and failures. (3) Decision Tree (DT), which takes the features of degradation to make the prediction. The performance comparisons are presented in Table 5. We observe that the behavior of the naive model in TeaVar is not satisfactory, with quite low values of P and R. The performance is improved in the statistical model, showing the power of utilizing the degradation signal to make the prediction. Our NN model with carefully characterized critical features performs the best, achieving 81% precision and recall. The high prediction accuracy demonstrates that the NN model makes an accurate prediction of failures in most of cases. Figure 14 shows the distribution of prediction error of all links. Compared to TeaVar, PreTE exhibits a smaller prediction error. To evaluate the contribution of features in our NN model, we conduct quantitative experiments to analyze each feature and observe that the $fiber\ ID$ plays the most important role in failure prediction. This phenomenon is expected since the likelihood of fiber failure upon the emergence of a degradation signal depends largely on the fiber itself. Please see Appendix A.6 for details. While it is possible to use a more accurate model, in the end, no model can capture all cuts with 100% accuracy. We will show our NN model does not lose much availability below.

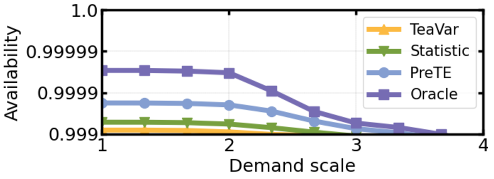
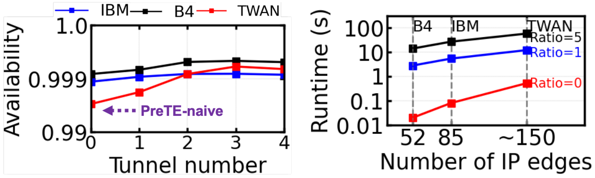


Figure 15: Performance comparisons at high availability levels on IBM topology.

Impact of prediction accuracy on availability. Given that the prediction accuracy varies significantly among different models, we then study the impact of prediction accuracy on network availability. Figure 15 depicts the performance comparisons at high availability levels on IBM topology. Here, we introduce the oracle which enables to make the prediction of fiber cuts with 100% accuracy. We observe that TeaVar's prediction approach performs the worst, as it does not rely on the degradation signal to predict failures. The availability reduces to 2 nines at the demand scale of 2.3. In contrast,

[4] P=TP/(TP+FP), R=TP/(TP+FN); where TP, FP and FN represent true positive, false positive and false negative respectively.



(a) Availability (b) Runtime

Figure 16: Impact of creating new tunnels.

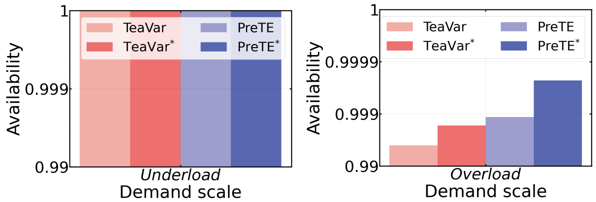
by utilizing the degradation signal, the network availability can maintain 3 nines at the same demand scale. PreTE further leverages the NN model to improve the availability, achieving a level close to 4 nines (99.98%) at the demand scale of 1. Note that the availability associated with oracular knowledge is 99.995%. PreTE maintains 3 nines even if the demand scale is 3.3 and downgrades to 2 nines at the demand scale of 3.7. This is quite close to the availability using the oracle knowledge. This phenomenon indicates that our proposed NN model does not lose much availability by making an accurate prediction of failures.

Impact of false positive. As PreTE is not able to achieve 100% prediction accuracy, we study the impact of the false positive, where tunnels are established to address failures but the fiber does not cut. We only need to update tens of tunnels for the potentially affected flow (see Figure 1(c) and Table 3). For example, only 13% of tunnels will be affected by the fiber cut on B4 topology, indicating that we only need to establish about 20 tunnels to address failures. It is important to note that a commercial router can always support tens of thousands of tunnels [19]. Thus, the resource overhead at the router can be considered negligible. Once the failure does not happen after a TE period of degradation, the tunnel will be updated to its original state to reduce the resource occupation.

6.4 Sensitivity Analysis

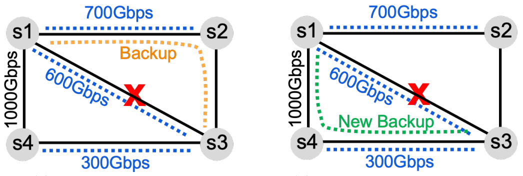
Impact of creating new tunnels on availability. When a tunnel is potentially affected, PreTE will establish new tunnels to absorb the network traffic. We then study the performance of the ratio of new tunnels when we observe that a tunnel is experiencing a degraded state. Figure 16(a) shows the network availability when the number of newly created tunnels changes. Here, the PreTE-naive means we do not establish any new tunnels for the tunnel experiencing a degraded state. We observe the performance gain of creating new tunnels at high availability levels. For example, the PreTE-naive only provides 2 nines (99.8%) on IBM topology. However, PreTE increases the availability to more than 3 nines (99.9%). This is because the newly established tunnels aggressively absorb the traffic to minimize the influence of failures. As the ratio increases, PreTE's availability gradually appears with fewer fluctuations, reflecting that all traffic suffering high failure probability has been perfectly rerouted, and continuing to scale the number of tunnels does not help much.

Impact of creating new tunnels on TE runtime. We then study the impact of creating new tunnels on the TE runtime. In Figure 16(b), we observe that the TE runtime is less than 1 second if we do not establish any tunnels for flows. However, if we establish one tunnel for every potentially affected tunnel (ratio=1), the TE runtime will be several seconds. For example, on B4 topology, it takes more than two seconds during the TE runtime, with most of the time spent on establishing new tunnels. This is attributed to the sequential establishment of a dozen tunnels, with each one taking hundreds of milliseconds to complete. However, creating a larger number of tunnels takes a long time, and it takes tens of seconds to create five tunnels for every potentially affected tunnel (ratio=5). To find a good balance between TE runtime and availability, the network operators should select an appropriate number of tunnels. As shown in Figure 5(a), most of the time interval between fiber degradation and fiber cut is more than 5 seconds. Hence, PreTE with a ratio equal to 1 is within the acceptable runtime to establish new tunnels before failures.



(a) Underload (scale=1) (b) Overload (scale=2.7)

Figure 17: Impact of uncertainty on B4.



(a) Traditional sytem (b) New system with PreTE

Figure 18: The performance gain of PreTE in production.

Impact of uncertainty. There are two important factors that will cause traffic dynamics in the network, i.e., workload uncertainty by traffic demand changes and capacity uncertainty by fiber cuts. These uncertainties will bring the traffic volume variation on tunnels. Please see Appendix A.7 for more details. We then study the flow availability if we can make the prediction and reduce the uncertainty in Figure 17. Here, PreTE reduces the capacity uncertainty by make the prediction of fiber cuts, as compared to TeaVar. TeaVar * and PreTE * reduce the workload uncertainty by make the prediction of future traffic demands, as compared to TeaVar and PreTE, respectively. We observe little improvement when we reduce the uncertainty when the network is underloaded (scale = 1). This phenomenon is reasonable because the network has enough bandwidth to tolerate these uncertainties. However, when the network is overloaded (scale=2.7), the performance gain by making the prediction of failures is larger than that of traffic demand prediction. For example, there is little performance gain in TeaVar * when we make the prediction of traffic demand. However, by comparing TeaVar * and PreTE *, the flow availability will raise from 99% to 999%, indicating a large performance gain by making the prediction of fiber cuts. This observation is reasonable since the changes in workload in a regular TE period are quite small; however the impact of fiber cut is large since it affects multiple IP links and cause several Tbps of IP capacity loss.

7 PRODUCTION CASE

Given that PreTE comprises multiple components, the deployment of the entire system into the production network presents significant challenges that any new component introduced to production will bring a risk to the system's reliability. Therefore, we are in the process of gradually deploying PreTE in production. At this stage, we have already deployed the failure prediction model and tunnel update module to facilitate the management of network traffic before failures, thus reducing packet loss.

Consider a production case in our network, shown in Figure 18(a), which includes a subset of our backbone topology with four sites as of 2025. Each IP link has a uniform bandwidth of 1000 Gbps. There are multiple tunnels on these links with tunnels $s1s2$, $s1s3$, and $s4s3$ allocated with network traffic of 700 Gbps, 600 Gbps, and 300 Gbps, respectively. Each of these tunnels will have an associated backup path; for instance, tunnel $s1s2s3$ is the backup path for tunnel $s1s3$. A fiber associated with IP link $s1s3$ evolves to a degraded state for tens of seconds and finally experiences failure. This failure will ultimately result in a complete loss of capacity for IP link $s1s3$. In the traditional system, when the router detects the link failure, it will switch the affected primary path $s1s2$ to the backup path $s1s2s3$ in a few seconds. However, the spare bandwidth on link $s1s2$ is insufficient for this additional traffic, leading to a continued packet loss. This issue will not be solved until the next TE period. In contrast, as for the new system with PreTE, when the controller detects a degradation signal that indicates a high failure probability, it will proactively calculate the optimal available backup tunnel, i.e., $s1s4s3$, for tunnel $s1s3$. When fiber cut happens and link $s1s3$ fails, the router can effectively switch the network traffic from tunnel $s1s2$ to tunnel $s1s4s3$, thereby avoiding any sustained packet loss.

8 DISCUSSION

Impact of data granularity. We take our one-second granularity data as the ground truth and study how the data granularity impacts on capturing these predictable fiber cuts. The predictable fiber cuts decrease to 2% with 5-minute granularity data (see Appendix A.8). Such coarse-grained data in traditional telemetry systems [13, 35] is hard to capture the network activities before failures. With a more advanced system to collect data at finer granularity, we believe that we can observe more predictive fiber cuts to generate efficient TE policies and improve network availability.

Impact of predictable fiber cuts on availability. To evaluate the ratio of predictable fiber cuts on availability, we study if more predictable fiber cuts will improve the network availability (see Appendix A.9). PreTE leverages 25% predictable fiber cuts to achieve a promising result with demand scaled by 3.3× at 99.9% availability. With all of the fiber cuts being predictable, the network availability is at a high level even if the demand scale is large. This motivates us to observe more optical indicators such as polarization mode dispersion, chromatic dispersion [11] to improve the predictability of fiber cuts in future work.

Leveraging more efficient model. PreTE leverages a vanilla neural network model and several features to obtain promising prediction results. It is important to improve the prediction accuracy since higher prediction accuracy can lead to higher network availability (see Figure 15). We can improve the prediction accuracy in the following ways. On the one hand, we can introduce more external features (e.g., weather events) and internal features (e.g., age of fiber, fiber vendor) closely related to fiber cuts as well as key features during the degradation period. On the other hand, it is also beneficial to explore the design of an effective deep neural network model that can efficiently capture the fiber state change before failures.

9 RELATED WORK

TE in WANs. TE has been studied intensively in WANs to globally optimize routing and traffic allocations [2, 4, 6, 14, 17, 18, 21, 24, 26, 29, 31, 31, 36, 39, 41]. We divide existing TE solutions into two categories: reactive and proactive (see Appendix A.10). The reactive solutions such as NCFlow [2], Flexile [21], and Teal [39] incorporated a centralized controller and globally updated TE policies in reaction to failures. Though such centralized decision-making ensures high utilization of available network capacity, they are slow in reacting to link failures—affecting availability. The proactive approaches such as TeaVar [6], proactively leave residual capacity for each tunnel, enabling them to handle more traffic during failure. However, these approaches operated at much coarse granularity by assuming the failure probability of network links stays invariant over time. PreTE is a hybrid prediction-based TE system that uses degradation signals to better estimate failure probabilities and reactively establish new tunnels and make effective TE decisions to proactively accommodate link failures.

Optical and IP layer orchestration. There have been several proposals [22, 41] to bridge the gap between optical and IP layers. Singh et al. measured the SNR of links in a large backbone and proposed RADWAN to evaluate interactions between dynamic capacity links and IP layer traffic [33–35]. Zhong et al. utilized the reconfigurability in the optical layer to accommodate short-term traffic spikes [42] and restore from fiber cut events with optimized traffic allocations [41]. Eason et al. [9] modeled the traffic based on the Hose model and optimizes optical layer resources based on the benders decomposition. Miao et al. [30] proposed a novel flexible WAN infrastructure to provision cost-effective WAN capacity while ensuring resilience to optical failures. Although with the similar motivation of achieving optical and IP layer orchestration, PreTE is orthogonal to previous work. By studying fiber degradations in the optical layer, PreTE shows that some fiber cut events are predictable. Then PreTE establishes new tunnels and makes more effective TE decisions in the IP layer to improve network availability.

10 CONCLUSION

This paper present a novel traffic engineering system called PreTE to factor in the dynamic fiber cut probabilities directly into TE systems. At the core of the PreTE system, fiber degradation facilitates failure predictions and traffic tunnels to be proactively updated, followed by traffic allocation optimizations among updated tunnels. We evaluate PreTE using a production-level WAN testbed and large-scale simulations. The testbed evaluation quantifies PreTE's runtime to demonstrate the feasibility to implement in large-scale WANs. Our large-scale simulation results show that PreTE can support up to 2× more demand at the same level of availability as compared to existing TE schemes.

This work **does not** raise ethical issues.

ACKNOWLEDGMENT

We sincerely thank our shepherd Anja Feldmann and anonymous SIGCOMM reviewers for their insightful comments. We also thank the teams at Tencent for their advice on PreTE. We are also grateful to Quan Yuan, Kunling He, and Zhiquan Wang for their help on the experiments. Jilong Wang was supported in part by the National Key Research and Development Program of China under Grant No. 2020YFE0200500. Jilong Wang is the corresponding author.

REFERENCES

[1] [n. d.]. Gurobi optimizer. http://www.gurobi.com.

[2] Firas Abuzaid, Srikanth Kandula, Behnaz Arzani, Ishai Menache, Matei Zaharia, and Peter Bailis. 2021. Contracting Wide-area Network Topologies to Solve Flow Problems Quickly. In *18th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 21)*. 175–200.

[3] Satyajeet Singh Ahuja, Varun Gupta, Vinayak Dangui, Soshant Bali, Abishek Gopalan, Hao Zhong, Petr Lapukhov, Yiting Xia, and Ying Zhang. 2021. Capacity-efficient and uncertainty-resilient backbone network planning with hose. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*. 547–559.

[4] Abd AlRhman AlQiam, Yuanjun Yao, Zhaodong Wang, Satyajeet Singh Ahuja, Ying Zhang, Sanjay G Rao, Bruno Ribeiro, and Mohit Tawarmalani. 2024. Transferable Neural WAN TE for Changing Topologies. In *Proceedings of the ACM SIGCOMM 2024 Conference*. 86–102.

[5] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. 2017. Julia: A fresh approach to numerical computing. *SIAM review* 59, 1 (2017), 65–98.

[6] Jeremy Bogle, Nikhil Bhatia, Manya Ghobadi, Ishai Menache, Nikolaj Bjørner, Asaf Valadarsky, and Michael Schapira. 2019. TEAVAR: striking the right utilization-availability balance in WAN traffic engineering. In *Proceedings of the ACM Special Interest Group on Data Communication*. 29–43.

[7] Marco Chiesa, Guy Kindler, and Michael Schapira. 2016. Traffic engineering with equal-cost-multipath: An algorithmic perspective. *IEEE/ACM Transactions on Networking* 25, 2 (2016), 779–792.

[8] Marek Denis, Yuanjun Yao, Ashley Hatch, Qin Zhang, Chiun Lin Lim, Shuqiang Zhang, Kyle Sugrue, Henry Kwok, Mikel Jimenez Fernandez, Petr Lapukhov, et al. 2023. Ebb: Reliable and evolvable express backbone network in meta. In *Proceedings of the ACM SIGCOMM 2023 Conference*. 346–359.

[9] John P Eason, Xueqi He, Richard Cziva, Max Noormohammadpour, Srivatsan Balasubramanian, Satyajeet Singh Ahuja, and Biao Lu. 2023. Hose-based cross-layer backbone network design with Benders decomposition. In *Proceedings of the ACM SIGCOMM 2023 Conference*. 333–345.

[10] Rob Enns, Martin Bjorklund, Juergen Schoenwaelder, and Andy Bierman. 2011. Network configuration protocol (NETCONF). (2011).

[11] Robert J Feuerstein. 2005. Field measurements of deployed fiber. In *National Fiber Optic Engineers Conference*. Optica Publishing Group, NThC4.

[12] Arthur M Geoffrion. 1972. Generalized benders decomposition. *Journal of optimization theory and applications* 10, 4 (1972), 237–260.

[13] Monia Ghobadi and Ratul Mahajan. 2016. Optical layer failures in a large backbone. In *Proceedings of the 2016 Internet Measurement Conference*. 461–467.

[14] Fei Gui, Songtao Wang, Dan Li, Li Chen, Kaihui Gao, Congcong Min, and Yi Wang. 2024. RedTE: Mitigating subsecond traffic bursts with real-time and distributed traffic engineering. In *Proceedings of the ACM SIGCOMM 2024 Conference*. 71–85.

[15] H3C. 2012. Bulk Interface Commands. https://www.h3c.com/en/d_201211/761369_294551_0.htm.

[16] Martin T Hagan, Howard B Demuth, and Mark Beale. 1997. *Neural network design*. PWS Publishing Co.

[17] Chi-Yao Hong, Srikanth Kandula, Ratul Mahajan, Ming Zhang, Vijay Gill, Mohan Nanduri, and Roger Wattenhofer. 2013. Achieving high utilization with software-driven WAN. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*. 15–26.

[18] Chi-Yao Hong, Subhasree Mandal, Mohammad Al-Fares, Min Zhu, Richard Alimi, Chandan Bhagat, Sourabh Jain, Jay Kaimal, Shiyu Liang, Kirill Mendelev, et al. 2018. B4 and after: managing hierarchy, partitioning, and asymmetry for availability and scale in google's software-defined WAN. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*. 74–87.

[19] Huawei. 2023. NE9000. https://support.huawei.com/enterprise/en/doc/EDOC1100278773/6d34bd5d/mpls.

[20] Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer, Junlan Zhou, Min Zhu, et al. 2013. B4: Experience with a globally-deployed software defined WAN. *ACM SIGCOMM Computer Communication Review* 43, 4 (2013), 3–14.

[21] Chuan Jiang, Zixuan Li, Sanjay Rao, and Mohit Tawarmalani. 2022. Flexile: meeting bandwidth objectives almost always. In *Proceedings of the 18th International Conference on emerging Networking EXperiments and Technologies*. 110–125.

[22] Xin Jin, Yiran Li, Da Wei, Siming Li, Jie Gao, Lei Xu, Guangzhi Li, Wei Xu, and Jennifer Rexford. 2016. Optimizing bulk transfers with software-defined optical WAN. In *Proceedings of the 2016 Conference of the ACM Special Interest Group on Data Communication*. 87–100.

[23] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[24] Praveen Kumar, Yang Yuan, Chris Yu, Nate Foster, Robert Kleinberg, Petr Lapukhov, Chiun Lin Lim, and Robert Soulé. 2018. Semi-oblivious traffic engineering: The road not taken. In *15th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 18)*. 157–170.

[25] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436–444.

[26] Hongqiang Harry Liu, Srikanth Kandula, Ratul Mahajan, Ming Zhang, and David Gelernter. 2014. Traffic engineering with forward fault correction. In *Proceedings of the 2014 ACM Conference on SIGCOMM*. 527–538.

[27] Hongzi Mao, Ravi Netravali, and Mohammad Alizadeh. 2017. Neural adaptive video streaming with pensieve. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*. 197–210.

[28] Congcong Miao, Minggang Chen, Arpit Gupta, Zili Meng, Lianjin Ye, Jingyu Xiao, Jie Chen, Zekun He, Xulong Luo, Jilong Wang, et al. 2022. Detecting ephemeral optical events with OpTel. In *19th USENIX Symposium on Networked Systems Design and Implementation*. 339–353.

[29] Congcong Miao, Zhizhen Zhong, Yunming Xiao, Feng Yang, Senkuo Zhang, Yinan Jiang, Zizhuo Bai, Chaodong Lu, Jingyi Geng, Zekun He, et al. 2024. MegaTE: Extending WAN Traffic Engineering to Millions of Endpoints in Virtualized Cloud. In *Proceedings of the ACM SIGCOMM 2024 Conference*. 103–116.

[30] Congcong Miao, Zhizhen Zhong, Ying Zhang, Kunling He, Fangchao Li, Minggang Chen, Yiren Zhao, Xiang Li, Zekun He, Xianneng Zou, et al. 2023. FlexWAN: Software Hardware Co-design for Cost-Effective and Resilient Optical Backbones. In *Proceedings of the ACM SIGCOMM 2023 Conference*. 319–332.

[31] Yarin Perry, Felipe Vieira Frujeri, Chaim Hoch, Srikanth Kandula, Ishai Menache, Michael Schapira, and Aviv Tamar. 2023. {DOTE}: Rethinking (Predictive){WAN} Traffic Engineering. In *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*. 1557–1581.

[32] Jeremy F Shapiro. 1976. Sensitivity analysis in integer programming. (1976).

[33] Rachee Singh, Nikolaj Bjorner, Sharon Shoham, Yawei Yin, John Arnold, and Jamie Gaudette. 2021. Cost-effective capacity provisioning in wide area networks with Shoofly. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*. 534–546.

[34] Rachee Singh, Monia Ghobadi, Klaus-Tycho Foerster, Mark Filer, and Phillipa Gill. 2017. Run, walk, crawl: Towards dynamic link capacities. In *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*. 143–149.

[35] Rachee Singh, Manya Ghobadi, Klaus-Tycho Foerster, Mark Filer, and Phillipa Gill. 2018. RADWAN: rate adaptive wide area network. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*. 547–560.

[36] Martin Suchara, Dahai Xu, Robert Doverspike, David Johnson, and Jennifer Rexford. 2011. Network architecture for joint failure recovery and traffic engineering. *ACM SIGMETRICS Performance Evaluation Review* 39, 1 (2011), 97–108.

[37] David Wetherall, Abdul Kabbani, Van Jacobson, Jim Winget, Yuchung Cheng, Charles B Morrey III, Uma Moravapalle, Phillipa Gill, Steven Knight, and Amin Vahdat. 2023. Improving Network Availability with Protective ReRoute. In *Proceedings of the ACM SIGCOMM 2023 Conference*. 684–695.

[38] Yiting Xia, Ying Zhang, Zhizhen Zhong, Guanqing Yan, Chiunlin Lim, Satyajeet Singh Ahuja, Soshant Bali, Alexander Nikolaidis, Kimia Ghobadi, and Manya Ghobadi. 2021. A Social Network Under Social Distancing: Risk-Driven Backbone Management During COVID-19 and Beyond.. In *NSDI*. 217–231.

[39] Zhiying Xu, Francis Y Yan, Rachee Singh, Justin T Chiu, Alexander M Rush, and Minlan Yu. 2023. Teal: Learning-Accelerated Optimization of WAN Traffic Engineering. In *Proceedings of the ACM SIGCOMM 2023 Conference*. 378–393.

[40] Ying Zhang, Nathan Hu, Carl Verge, and Scott O'Brien. 2022. Cross-layer diagnosis of optical backbone failures. In *Proceedings of the 22nd ACM Internet Measurement Conference*. 419–432.

[41] Zhizhen Zhong, Manya Ghobadi, Alaa Khaddaj, Jonathan Leach, Yiting Xia, and Ying Zhang. 2021. ARROW: restoration-aware traffic engineering. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*. 560–579.

[42] Zhizhen Zhong, Nan Hua, Massimo Tornatore, Jialong Li, Yanhe Li, Xiaoping Zheng, and Biswanath Mukherjee. 2019. Provisioning short-term traffic fluctuations in elastic optical networks. *IEEE/ACM Transactions on Networking* 27, 4 (2019), 1460–1473.

Appendices are supporting material that has not been peer-reviewed.

A APPENDIX

A.1 Conformation with hypothesis test

Epoch Number	#Degradation	#No degradation	Total
#Failure	1	2.6	3.6
#No failure	1.5	6516.7	6518.2
Total	2.5	6519.3	6521.8

Table 6: The normalized number of events.

We firstly discretize the one-year timeline into fifteen-minute epochs and count the total number of fiber cuts (i.e., failures) and fiber degradations, as shown in Table 6. We set the null hypothesis to be: fiber cuts are not related to fiber degradations. We apply a chi-square test and calculate the P-value, which is less than 10^{-50}. Considering a p-value threshold of 0.01, the null hypothesis is rejected, indicating that fiber cuts and degradations are related in a statistically significant manner.

We then study the statistical number of fiber failures and degradations that the null hypothesis cannot be rejected. The detailed number is shown in Table 7. We observe that the number of epochs that failures and degradations fall into the same epoch should be extremely low (i.e., 1.2) if they do not have the relations.

Number	#Degradation	#No degradation	Total
#Failure	1.2	3151.8	3153
#No failure	2144.8	5655630.2	5657775
Total	2146	5658782	5660928

Table 7: The number of events with the null hypothesis not being rejected.

A.2 Neural network implementation

The neural network (NN) algorithm is shown in Figure 9. For the input variables, the variables *degree*, *gradient*, *fluctuation*, and *length* are scaled into [0,1] using Min-Max normalization method, i.e., $x^* = (x - MIN)/(MAX - MIN)$. The variables *time*, *region* and *fiber ID* are encoded into binary vectors with one-hot encoding. To reduce the curse of dimensionality, we represent variables *region* and *fiber ID* with a low-dimensional vector with a transformation matrix to capture precise semantic information, namely variable embedding. The hidden layer uses 64 neurons to aggregate these inputs, and another hidden layer that uses 2 neurons that act as a decoder to project raw values into a vector with two dimensions. The output of NN is a two-dimensional vector $\vec{y}$. We apply a softmax layer such that the output probability distribution $\vec{p}$ is normalized, i.e., $p_0 + p_1 = 1$ where p_0 and p_1 represent the estimated probability of normal and failure respectively. For a binary classification problem, the loss function is regarded as a negative log-likelihood of ground truth: $\mathcal{L}(\theta) = -d_i log(p_i)$, where d_i is the mask of ground truth; θ is set of trainable parameters.

The learning rate is configured to be 10^{-3}. We keep all these hyper-parameters fixed throughout our experiments. To avoid the overfitting problem, we use two regularization techniques, we use L2 regularization set to 0.0002 for all trainable parameters. While some tuning is useful, we found that the NN algorithm performs well for a wide range of hyper-parameter values. Thus we did not use sophisticated hyper-parameter tuning methods. Our neural network model is implemented on Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz using PyTorch and trained offline with the Adam algorithm [23]. We use the first 80% of each fiber's degradation signals as training data and the remaining 20% of each fiber's degradation signals are used as testing data for evaluating the model's performance.

A.3 Proof of Theorem 4.1

To prove Theorem 4.1, given observed time epochs T, the number of fiber cuts and fiber degradations is Tp_i and Tp_d respectively. As α of fiber cuts are predictable, the number of predictable and unpredictable fiber cuts is αTp_i and $(1-\alpha)Tp_i$ respectively. Since these unpredictable fiber cuts follow a geometric distribution, the failure probability is represented as:

$$\frac{(1-\alpha)p_i T}{T(1-p_d)} \approx (1-\alpha)p_i$$

when we do not observe the fiber degradation.

A.4 Bender's Decomposition

The whole procedure of the algorithm is stated as follows, as well as pseudocode in Algorithm 2.

Initialization. A good initialization of $\delta_{f,q}$ makes the algorithm closer to the real minimizer at the beginning, such that the algorithm achieves optimal solutions with few iterations (line 2). Randomized initialization of $\delta_{f,q}$ makes the algorithm mostly being far away from the real minimizer. An alternative optimized choice is to initialize $\delta_{f,q} = 1$, representing all scenarios are selected for all flows. Such initialization directly satisfies the constraint (5). We define UB and LB of Φ to 1 and 0 respectively since the normalized loss of flows should be between 0 and 1 (line 3). Furthermore, we set the optimality cut set $C = \emptyset$ in the initial step (line 4).

Subproblem (SP). After the set of scenarios for each flow is determined by fixing $\delta_{f,q}$, the subproblem is constructed.

$$(\mathbf{SP}) \min_{a,l,\Phi} \Phi \quad \mathbf{s.t.} : (3,4,6,8) \tag{9}$$

Note, the subproblem is a pure LP problem with the integer variable $\delta_{f,q}$ being fixed. To solve the SP in parallel, we turn the $\delta_{f,q}$ in the constraint (6) into the objective function. We obtain different objective functions as $\delta_{f,q}$ changes, however with the same constraints. The sizes of the SP to be solved are independent of the number of $\delta_{f,q}$ to be addressed. We derive the dual problem of SP as follows.

$$(\mathbf{D}) \max \sum_{f\in F, q\in Q} [(\delta_{fq}-1)w_{fq} + d_f v_{fq} - o_{fq}] - \sum_{e\in E} c_e u_e \tag{10}$$

The detailed process of obtaining the dual problem is shown in Appendix A.5.

After solving D problem, a set of constants π =($\hat{w}$, $\hat{v}$, $\hat{o}$, $\hat{u}$) are obtained as well as traffic allocation $a_{f,t}$ (line 6). Then we calculate a new upper bound $UB^* = \sum_{f\in F, q\in Q}[(\delta_{fq}-1)\hat{w}_{fq} + d_f\hat{v}_{fq} - \hat{o}_{fq}] - \sum_{e\in E} c_e\hat{u}_e$ (line 7). Since the optimal value of the D problem equals to the optimal value of the SP problem, the fact that the fixed value $\delta_{f,q}$ makes the value UB^* implies that the target Φ of the SP, as well as the master problem is upper bounded by UB^*. Therefore, we derive an optimality cut c:

$$\Phi \geq \sum_{f\in F, q\in Q} [(\delta_{fq}-1)w_{fq} + d_f\hat{v}_{fq} - \hat{o}_{fq}] - \sum_{e\in E} c_e\hat{u}_e \tag{11}$$

Then, the new optimality cut c is further added into the optimality cut set C (line 8).

Master problem (MP). After updating the constraint set C, the master problem can find a better solution of the integer variable δ. Note, the MP only needs to derive a feasible variable δ with an

Algorithm 2 Bender's decomposition algorithm

```
1:  procedure CALCULATE TRAFFIC ALLOCATION $a_{f,t}$ AND LOSS $\Phi$
      ▷ Input: $\varepsilon$: a threshold for coverage checking
      ▷ Output: $a_{f,t}$ : allocated bandwidth on tunnel $t$ for flow $f$
      ▷ Output: $\Phi$ : maximum loss across all flows
2:      Initialize scenarios $\delta_{f,q} \leftarrow 1$, ∀ f,q;
3:      Initialize lower bound $LB \leftarrow 0$, upper bound $UB \leftarrow 1$;
4:      Initialize optimality cut set $C \leftarrow \emptyset$;
5:      while UB-LB > $\varepsilon$ do
            ▷ Step 1: solve the subproblem
6:          $a_{f,t}$,$\pi$ ← Slove SP with the fixed $\delta_{f,q}$;
7:          $UB^*$ ← Calculate upper bound with $\delta_{f,q}$ and $\pi$;
8:          C.add($c$);
            ▷ Step 2: solve the master problem
9:          $\delta_{f,q}$ ← Slove MP with constraints (5, 7 ∪ C);
10:         $LB^*$, $\Phi$ ← Calculate lower bound by (14);
            ▷ Step 3: Bound update
11:         $UB = \min(UB, UB^*)$;
12:         $LB = \max(LB, LB^*)$;
13:     return $a_{f,t}$, $\Phi$
```

underestimate of the optimal loss. Thus, the master problem which is related to a small scale binary variable can be solved with slack variables [32].

$$(\mathbf{MP}) \min_{\delta,\Phi} \Phi \quad \mathbf{s.t.} : (6, 8) \tag{12}$$

$$\forall c \in C : \Phi \geq c(\delta) \tag{13}$$

After solving the **MP**, we obtain a new binary variable $\delta_{f,q}$ (line 9), as well as a lower bound LB^* by fixed $\delta_{f,q}$ (line 10).

Coverage checking. Each iteration generates a new upper bound UB^* and lower bound LB^* which are used to update the bound of the original problem (line 11-12). We compare the LB and UP each time after updating the bounds and terminate the algorithm if $UB - LB \leq \epsilon$ (line 5). Finally, the maximum flow loss Φ and traffic allocation $a_{f,t}$ are returned.

A.5 Dual of subproblem

With the complicating variable $\delta_{f,q}$ being fixed, we obtain the subproblem. The details of Eqn. (9) is shown below:

$$(\mathbf{SP}) \min_{a,l,\Phi} : \quad \Phi$$

Subject to :

$$\forall f, q : \sum_{t \in T_{f,q}} a_{f,t} + \sum_{t \in T^s_{f,q}} a_{f,t} \geq (1 - l_{f,q}) d_f$$

$$\forall e : \sum_{f \in F, t \in T_f} a_{f,t} L(t,e) + \sum_{f \in F, t \in T^s_f} a_{f,t} L(t,e) \leq c_e$$

$$\forall f, q : \Phi \geq l_{f,q} - 1 + \delta_{f,q}$$

$$\forall f, q, t : 0 \leq l_{f,q} \leq 1, 0 \leq a_{f,t}$$

We could turn the $\delta_{f,q}$ in the constraint (6) into the objective function. Therefore we could get different objective function as the $\delta_{f,q}$ changes, however with the same constraints. The sizes of the **SP** to be solved are independent of the number of $\delta_{f,q}$ addressed and we could solve **SP** in parallel. We firstly rewrite the subproblem as follows:

$$(\mathbf{SP'}) \min_{a,l,\Phi} : \quad 1 \cdot \Phi + \sum_{f \in F, q \in Q} 0 \cdot l_{fq} + \sum_{f \in F, t \in T_f} 0 \cdot a_{ft}$$

Subject to :

$$\forall f, q : \sum_{t \in T_{f,q}} a_{f,t} + \sum_{t \in T^s_{f,q}} a_{f,t} + + l_{f,q} d_f \geq d_f$$

$$\forall f, q : - \sum_{f \in F, t \in T_f} a_{f,t} L(t,e) - \sum_{f \in F, t \in T^s_f} a_{f,t} L(t,e) \geq -c_e$$

$$\forall f, q : \Phi - l_{f,q} \geq \delta_{f,q} - 1$$

$$\forall f, t : a_{ft} \geq 0$$

$$\forall f, q : -l_{fq} \geq -1$$

$$\forall f, q : l_{fq} \geq 0$$

Notice that the right-hand side of the constraints contains the constants δ_{fq}. To simplify the calculation, we derive the dual problem of (S'). The dual problem is as follows:

$$(\mathbf{D}) \max_{w,v,o,u} : \sum_{f \in F, q \in Q} (\delta_{fq} - 1) w_{fq} + \sum_{f \in F, q \in Q} d_f v_{fq} - \sum_{f \in F, q \in Q} o_{fq} - \sum_{e \in E} c_e u_e$$

Subject to :

$$: \sum_{f \in F, q \in Q} w_{fq} \leq 1$$

$$\forall f, q : -w_{fq} + d_f v_{fq} - o_{fq} \leq 0$$

$$\forall f, t : \sum_{q \in Q} X(t,q) v_{f,q} - \sum_{e \in E} L(t,e) u_e \leq 0$$

$$\forall f, q, e : w_{fq} \geq 0, \; v_{fq} \geq 0, \; u_e \geq 0, \; o_{fq} \geq 0$$

A.6 Quantitative Comparison with variants of neural network model.

To evaluate the contribution of critical features (e.g., time, degree, gradient, fluctuation) and intrinsic features (e.g., region, vendor) that are used in in our proposed NN model, we conduct quantitative experiments to analyze each feature while keeping all hyperparameters at the optimal settings. The experiment results are shown in Table 8. Here, NN w/o x indicates that the input to the neural network model excludes the feature x and NN-*all* means all features are considered in the neural network model. *Accuracy* is the straightforward performance indicator, which is the ratio of correct predictions, while $F1$ is the harmonic mean of precision P and recall R, which can be regarded as an overall performance indicator. We observe that NN-*all* outperforms all other approaches by achieving highest values of all metrics This phenomenon points out that these features are always beneficial. Another important observation is that NN w/o *fiber ID* performs the worst with the lowest F1 and accuracy, indicating that the fiber ID plays the most important role in failure prediction. This phenomenon is expected since the likelihood of fiber failure upon the emergence of a degradation signal depends largely on the fiber itself.

Method	P	R	F1	Accuracy
NN w/o *time*	0.76	0.76	0.76	0.76
NN w/o *gradient*	0.78	0.74	0.76	0.76
NN w/o *degree*	0.78	0.81	0.79	0.79
NN w/o *fluctuation*	0.77	0.79	0.79	0.78
NN w/o *region*	0.76	0.78	0.78	0.77
NN w/o *fiber ID*	0.58	0.81	0.68	0.61
NN w/o *vendor*	0.79	0.76	0.77	0.78
NN-*all*	0.81	0.81	0.81	0.81

Table 8: Variants of NN model with different features.

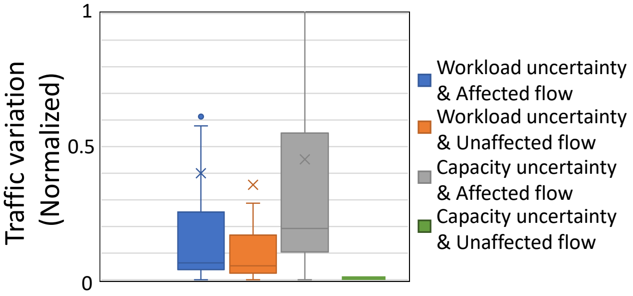


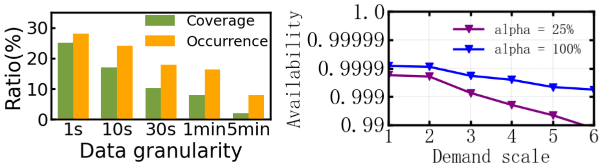
Figure 19: Traffic variation caused by workload uncertainty and capacity uncertainty.

A.7 Impact of uncertainty on traffic variation.

We study the impact of these uncertainties on two classes of flows separately, i.e., affected flow and unaffected flow, depending on whether the flow is affected by the failure or not. Figure 19 illustrates the traffic variation caused by these uncertainties. Here, the traffic variation refers to the fluctuations in traffic volume within the tunnel, which are influenced by the fluctuations in adjacent traffic demand (workload uncertainty) or by the conditions before and after failures (capacity uncertainty). For these flows affected by the failure, we observe that the traffic variation by workload uncertainty is small (see workload uncertainty & affected flow in the Figure 19). This is largely due to the small workload changes in a regular TE period. In contrast, the traffic variation due to the capacity uncertainty is much larger. We attribute this phenomenon to two reasons. On the one hand, a single fiber cut affects multiple IP links, causing several Tbps of IP capacity loss and impacting a lot of flows and tunnels. On the other hand, since the proactive TE approach only locally updates the rate adaptation policy at the affected endpoints, the traffic variation on the corresponding tunnels will be very large. For the unaffected flows, we still observe a small traffic variation by the workload uncertainty. The values are quite close to the affected flows by workload uncertainty.

A.8 Impact of data granularity.

We study how the data granularity impacts the capture of these predictable fiber cuts. We take our one-second granularity data as the ground truth and simulate the detection of events with different granularities. We introduce *coverage* ratio as the ratio of predictable cuts to total cuts and, and *occurrence* ratio as the ratio of predictable cuts to total degradations. Figure 20(a) shows that the coverage ratio is 25% with one-second granularity data and decreases to 2% with 5-minute granularity data. Such coarse-grained data in traditional telemetry systems [13, 35, 40] is hard to capture the network activities before failures. On the other hand, the occurrence ratio is less than 10% with 5-minute granularity data, which hinders establishing an accurate model to predict failures. With a more advanced system to collect data at finer granularity, we believe we can observe more predictive fiber cuts and an accurate model to generate efficient TE policies and improve network availability.



(a) Data granularity (b) Availability

Figure 20: Impact of data granularity on availability.

A.9 Impact of predictable fiber cuts α.

We change the coverage ratio and obtain availability with different demand scales. Figure 20(b) shows the network is at a high availability level if 100% of fiber cuts are predictable. The availability is 3 nines even if the demand is scaled by 6×. PreTE relying on 25% predictable fiber cuts can achieve promising results when the demand is scaled by 3.3×.

A.10 Comparing PreTE with existing TE system

TE systems	Pre-failure TE policy				Failure Reactions	
	Degradation aware	Failure aware	Probabilistic failures	Tunnel updates	Reaction method	Reaction speed
NCFlow[2]	No	No	No	No	Reactive	Seconds
Flexile[21]	No	Yes	Fixed	No	Reactive	Seconds
ECMP[7]	No	No	No	No	-	ms
SMORE[24]	No	No	No	No	-	ms
FFC[26]	No	Yes	No	No	Proactive	ms
TeaVar[6]	No	Yes	Fixed	No	Proactive	ms
ARROW[41]	No	Yes	Fixed	No	Proactive	Seconds
Teal [39]	No	Yes	Fixed	No	Reactive	Seconds
DOTE [31]	No	Yes	Fixed	No	Proactive	ms
PreTE (Ours)	Yes	Yes	Dynamic	Yes	Proactive	ms

Table 9: Comparing PreTE with existing TE systems.