A Social Network Under Social Distancing: Risk-Driven Backbone Management During COVID-19 and Beyond

Yiting Xia, *MPI-INF and Facebook;* Ying Zhang, *Facebook;* Zhizhen Zhong, *MIT and Facebook;* Guanqing Yan, Chiun Lin Lim, Satyajeet Singh Ahuja, Soshant Bali, and Alexander Nikolaidis, *Facebook;* Kimia Ghobadi, *Johns Hopkins University;* Manya Ghobadi, *MIT*

https://www.usenix.org/conference/nsdi21/presentation/xia

This paper is included in the Proceedings of the 18th USENIX Symposium on Networked Systems Design and Implementation.

April 12–14, 2021

978-1-939133-21-2

Open access to the Proceedings of the 18th USENIX Symposium on Networked Systems Design and Implementation is sponsored by

A Social Network Under Social Distancing: Risk-Driven Backbone Management During COVID-19 and Beyond

Yiting Xia[*§] *Ying Zhang*[§] *Zhizhen Zhong*[†§] *Guanqing Yan*[§] *Chiun Lin Lim*[§]
Satyajeet Singh Ahuja[§] *Soshant Bali*[§] *Alexander Nikolaidis*[§] *Kimia Ghobadi*[‡] *Manya Ghobadi*[†]
[*] *MPI-INF* [§] *Facebook* [‡] *Johns Hopkins University* [†] *MIT*

Abstract

As the COVID-19 pandemic reshapes our social landscape, its lessons have far-reaching implications on how online service providers manage their infrastructure to mitigate risks. This paper presents Facebook's risk-driven backbone management strategy to ensure high service performance throughout the COVID-19 pandemic. We describe Risk Simulation System (RSS), a production system that identifies possible failures and quantifies their potential severity with a set of metrics for network risk. With a year-long risk measurement from RSS we show that our backbone resiliently withstood the COVID-19 stress test, achieving high service availability and low route dilation while efficiently handling traffic surges. We also share our operational practices to mitigate risk throughout the pandemic.

Our findings give insights to further improve risk-driven network management. We argue for incorporating short-term failure statistics in modeling failures. Common failure prediction models based on long-term modeling achieve stable output at the cost of assigning low significance to unique short-term events of extreme importance such as COVID-19. Furthermore, we advocate augmenting network management techniques with non-networking signals. We support this by identifying and analyzing the correlation between network traffic and human mobility.

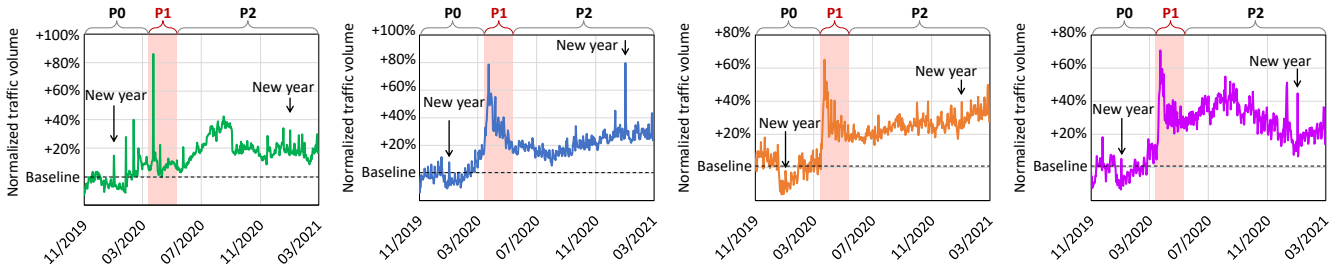
1 Introduction

COVID-19 fundamentally reshaped societal norms and human interactions by forcing most social activities to move online. The global network infrastructure was subjected to an unprecedented stress test as work, entertainment and education all had to be conducted via digital connections [37]. Over the past year, the networking community aimed to answer two fundamental questions about the impact of COVID-19 on different network environments [7, 15, 28]. First, how well has the current network infrastructure withstood the COVID-19 stress test? Second, how should the network infrastructure evolve to support a post-pandemic era likely to be permanently remodeled by the social distancing experience?

This paper supplements the recent COVID-19-centric research by sharing Facebook's experience emerging from the risk-driven backbone management strategy. Our work has two unique angles: the focus on the backbone network of a global online service provider and the use of network risk to quantify the robustness of the network infrastructure under adverse conditions. This study enriches previous observations made in different network environments, including the Internet [15], edge networks [7], and mobile networks [28]. Furthermore, it is a significant departure from prior work which uses only traffic measurement to quantify the impact of social events on the network infrastructure.

Our risk-driven backbone management is based on the fact that failures and disasters happen frequently and the backbone network should be equipped with sufficient protection capacity to mitigate the effects. Particularly, Facebook's backbone connects hundreds of Point-of-Presence (PoP) sites and tens of Data Center (DC) regions. At this scale, failures such as fiber cuts, router misconfigurations, and power outages happen on a daily basis [20], causing traffic congestion, packet loss, and latency increase which, in turn, negatively impact the network's availability and service-level agreements [8, 21, 27]. *Network risk* is an effective means to capture the impacts of potential failures in the network, before they actually occur, which is critical for identifying operational pain-points for long-term deployment planning, mid-term capacity augmentation, and short-term health monitoring.

This paper describes our Risk Simulation System (RSS), which performs comprehensive "what-if" analyses of network risk through traffic simulations under plausible failure scenarios. RSS has been in production for years. We introduce RSS in detail, showing key design decisions and engineering efforts to optimize the system over time. Specifically, we propose a set of risk metrics (demand loss, availability, latency stretch) to quantify impacts of potential failures from different aspects. Further, we introduce a high-fidelity failure model based on failure records from different data sources



(a) Asia (b) Europe (c) North America (d) South America

Figure 1: User-facing (DC-to-PoP) traffic traversing the global backbone.

and failure types. To scale our system, we provide different simulation granularities that trade failure count for simulation accuracy. Moreover, we discuss several techniques to accelerate computation such as system parallelization, routing simplification, and reduction of failure scenarios.

We conduct a year-long analysis of network risk using RSS, and in what follows, we share our operational experience of keeping risk at bay during the COVID-19 pandemic. Our risk analysis demonstrates our backbone network remained robust under the COVID-19 traffic surges. Although risk increased with traffic, even the most heavily affected class of service still achieved four 9s of availability and its flows only experienced 2.12% longer paths on average. We further discuss capacity enhancement and quality of service downgrade as two effective measures to reduce network risk.

Finally, we use case studies to show unusual network conditions caused by social distancing have challenged fundamental assumptions of the traditional network design, and we share our insights on future directions of risk-driven backbone management. We observe a large variation of optical and IP-layer failures triggered by changes of human activities. We thus suggest failure modeling to be more responsive to short-term failure statistics and discuss the tradeoff between model stability and agility for accurate failure predictions. We also identify the limitation of standard network management that only considers in-network signals and is blind to social impacts to the network. We find a negative correlation between traffic volume and population mobility rate during social distancing, and use it as an example to show opportunities for improving network management with external non-networking signals.

Our risk metrics, failure model and risk simulation approach generalize beyond the initially envisioned backbone network scenario and are readily applicable to other network environments. We believe that risk-driven network management has the potential to become the standard approach to disaster prevention, monitoring and recovery. Towards this goal, we hope that our experience can inspire future solutions and spur broader adoption of risk-driven network management. This work does not raise any ethical issues. We preserved user privacy and anonymity throughout this study.

Phase	Start date	End date	# Days
Pre-COVID (P_0)	11/4/2019	3/15/2020	133
Shelter-in-Place (P_1)	3/16/2020	5/3/2020	49
Re-opening (P_2)	5/4/2020	2/28/2021	301

Table 1: Measurement phases in this paper.

2 Traffic Surges During COVID-19

Recent news reports and measurement studies suggest significant traffic surges globally during the COVID-19 pandemic [7, 10, 15, 24, 28]. As a major social media platform, Facebook witnessed higher user engagement under social distancing. In this section, we measure Facebook's user-facing traffic to motivate the need for a risk-driven backbone management system.

Throughout the paper, we categorize our measurement period into three time phases (P_0 to P_2) listed in Table 1. The first phase, *Pre-COVID* (P_0), is our baseline to capture the state of the network before the global shut-down due to the pandemic. The second phase, *Shelter-in-Place* (P_1), marks the period when the US and European countries started to introduce extreme COVID-19 regulations, such as border closures, flight reductions, and school closures. The third phase, *Re-opening* (P_2), represents the slow re-opening phase when strict shut-down orders were relaxed [2].

Significant traffic increase. Figure 1 plots the traffic volume from our Data Center (DC) regions to Point-of-Presence (PoP) sites in four geographical regions. The traffic volume is normalized against average traffic during the pre-COVID phase (P_0). The figure shows a significant traffic surge starting mid-March 2020 in all regions, matching the timeline of global social distancing. In particular, we measure a traffic surge of 86% in Asia, 78% in Europe, 65% in North America, and 70% in South America in the P_1 phase.

Beyond the New Year traffic spike. Traffic volume spikes are not unusual. Today's service providers consider well-known flash-crowd events, such as Cyber Monday, in their traffic modeling [45] and network operation planning [49]. However, as Figure 1 depicts, the COVID-19 traffic increase has two unique differences. First, the traffic peak during phase P_1 was substantially higher than that of New Year's Eve (31

December 2019). The peak volume was 1.62× the 2020 New Year's Eve in Asia, 1.65× in Europe, 1.68× in North America, and 1.61× in South America. Second, flash-crowd events are usually short-lived, but the traffic surges remained high for several weeks during the pandemic.

The above observations highlight the challenges posed by social distancing on large-scale network operations. Under high traffic load, operators need to answer a natural question: "is my network at risk?" This question motivates us to quantify network risk and use it as a guiding signal to drive network management.

3 Risk-Driven Backbone Management

Satisfying Service Level Objectives (SLOs) is the ultimate goal of network management. Risk analysis is an indispensable and effective means to guarantee SLO compliance under different failure scenarios. In this section, we dive into the details of RSS, our risk-driven backbone management framework. We begin with the description of Facebook's traffic classification and routing schemes (§3.1), followed by our definition of risk metrics that align with SLO requirements of different service classes (§3.2). Next, we describe our failure modeling technique (§3.3). Finally, we present the design and implementation of RSS, our risk simulation system (§3.4).

3.1 Traffic Classification and Routing

Quality of Service (QoS). Facebook classifies the backbone traffic into four service classes. In this paper, we refer to them as QoS classes 1 to 4, where class 1 is the highest priority. Different classes of service use different queue assignments and routing policies. Flows with higher priorities have greater availability guarantees and can tolerate more failures compared to those in lower priority classes. This is often realized by over-provisioning extensive backup paths for redundancy. QoS class 1 contains essential network control traffic including network signaling and routing protocol messages to manage our network gear; class 2 is for critical services including most of our user-facing traffic; class 3 is our default class for most internal applications; and class 4 is for heavy, bulk data transfers. To reduce operational costs, we constantly look for opportunities to move traffic into class 4.

Routing. Our backbone uses a centralized network controller to make routing and Traffic Engineering (TE) decisions [22]. The centralized controller implements different traffic allocation algorithms for different QoS classes. To minimize the latency experienced by flows in QoS classes 1 and 2, we use a Constrained Shortest Path First (CSPF) approach that provisions TE tunnels for these flows up to the physical capacity of the network links. Flows are assigned to paths with the smallest round trip latencies. The bandwidth for QoS class 2 is allocated after class 1, and we reserve headroom on each link for potential traffic bursts for these two classes. QoS classes 3 and 4 use a combination of K-Shortest Paths (KSP) and Multi-Commodity Flow (MCF) algorithms with the objective of minimizing the maximum link utilization in the network. We pre-assign TE tunnels for traffic flows between each router pair, then rely on a Linear Program (LP) to load-balance the traffic over all tunnels. Lower priority traffic uses the bandwidth left by higher priority traffic on each link. When failures bring certain links down, traffic is automatically re-distributed across remaining tunnels until the next TE execution where a new optimal traffic allocation is calculated.

Algorithm 1 Compute risk metrics for QoS class q

procedure CALCULATE DEMAND LOSS, AVAILABILITY AND LATENCY STRETCH FOR QOS CLASS q UNDER FAILURE SCENARIOS S
▷ **Input:** S: set of considered failure scenarios
▷ **Input:** T: set of Traffic Engineering tunnels on the IP topology G
▷ **Input:** $F^q = \{f\}$: set of flows in QoS class q
▷ **Input:** d_f: bandwidth demand of flow f
▷ **Output:** V^q: QoS class q's availability
▷ **Output:** $L_f^q = \{< L_f^{s,q}, s.probability >\}$: for flow f, a distribution of latency stretch $L_f^{s,q}$ per failure scenario and its failure probability
▷ **Output:** X^q: QoS class q's demand loss
Initialize flow f's availability $V_f^q = 1, \forall q, f$
Initialize flow f's demand loss in scenario s: $X_f^{s,q} = 0, \forall q, f$
▷ *Iterate on all failure scenarios in S*
for all $s \in S$ **do**
▷ *TE bandwidth allocation $b_f^{s,q}$ and per-tunnel split ratio $a_{f,t}^{s,q}$*
$\{b_f^{s,q}\}, \{a_{f,t}^{s,q}\} =$ **TrafficEngineering**(G, T, F^q, s)
for all $f \in F^q$ **do**
▷ *Flow f's bandwidth-weighted latency*
$l = (\sum_{t \in T_f} t.rtt \times a_{f,t}^{s,q}) / (\sum_{t \in T_f} a_{f,t}^{s,q})$
▷ *Flow f's latency stretch*
$L_f^{s,q} = l / \min_{t \in T_f} t.rtt$
L_f^q.append(<$L_f^{s,q}$, $s.probability$>)
if $b_f^{s,q} < d_f$ **then**
▷ *Flow f's demand loss*
$X_f^{s,q} = X_f^{s,q} + (d_f - b_f^{s,q})$
▷ *Flow f's availability reduction*
$V_f^q = V_f^q - s.probability$
$V^q = \min_{f \in F^q}(V_f^q)$
$X^q = \max_{s \in S}(\sum_{f \in F^q} X_f^{s,q})$
return V^q, L_f^q, X^q

3.2 Risk Metrics

This paper defines a set of key metrics to quantify a network's risk to potential failures. These risk metrics satisfy two requirements. First, they capture different aspects of failure events by quantifying the network's response from multiple dimensions. Second, since QoS classes have different levels of tolerance to failures, our risk metrics relate to QoS classes and reflect their SLO guarantees. As a result, we use the following metrics for each QoS class q:

(1) Demand Loss (X^q): For a failure scenario s, the demand loss is the total amount of lost traffic by all flows in q caused by s. The overall demand loss of QoS class q is the maximum, or worst-case, demand loss across all the considered failure

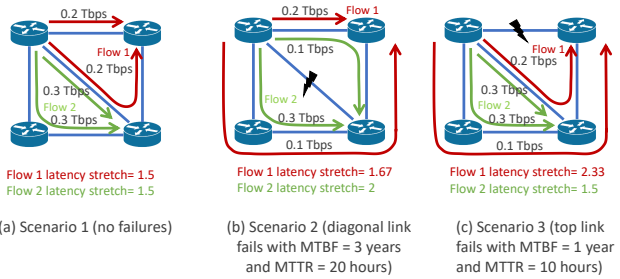


Figure 2: An example with two flows (green and red arrows) from the same QoS class q under three failure scenarios. Risk metrics computed by Algorithm 1 are as follows: worst-case demand loss (X^q) = 0.3 Tbps, worst-case availability (V^q) = 99.81%, flow 1's latency stretch (L_1^q) = {<1.5, 99.81%>, <1.67, 0.18%>, <2.33, 0.11%>}, and flow 2's latency stretch (L_2^q) = {<1.5, 99.81%>, <2, 0.18%>, <1.5, 0.11%>}.

scenarios $s \in S$.

(2) Availability ***(V^q)***: The percentage of time that a flow's demand is completely satisfied (100% admitted) across all failure scenarios reflects the availability of that flow. Similar to demand loss, we compute our availability metric as the lowest availability among all flows in QoS class q.

(3) Latency Stretch ***(L_f^q)***: For a failure scenario s, the latency stretch $L_f^{s,q}$ of flow f in QoS class q is the ratio of the average tunnel latency (weighted by the tunnel bandwidth assignments) divided by the shortest TE tunnel latency without failure. We use Round-Trip Time (RTT) as a proxy for tunnel latency but other metrics such as hop-count and fiber length can also be used. The overall latency stretch L_f^q across all failure scenarios is a distribution, represented as the latency stretch $L_f^{s,q}$ of each failure scenario associated with its time probability of occurrence.

Algorithm 1 describes how these metrics are calculated by RSS in detail. We first initialize each flow's availability and demand loss to 1 and 0, respectively (lines 2-3). For all considered failure scenarios $s \in S$, we execute the Traffic Engineering (TE) formulation to find the per-flow satisfied bandwidth $b_f^{s,q}$ and the per-tunnel traffic allocations $a_{f,t}^{s,q}$ for each flow (line 5). Then, we calculate the three risk metrics using the outputs of TE. Flow f's latency stretch is calculated as the bandwidth-weighted latency l divided by the minimum latency across all tunnels (lines 7-8). Flow f's demand loss is captured by the difference between satisfied bandwidth $b_f^{s,q}$ and flow's demand d_f (line 11). If the demand is not fully satisfied, availability is reduced by the probability of the failure scenario s (line 12). Finally, we select the availability of QoS class q to be the worst availability experienced by all flows $f \in F^q$ (line 13), and demand loss to be the maximum loss across all scenarios $s \in S$ (line 14).

Figure 2 shows an example of the risk metrics computed by Algorithm 1 for two flows from the same QoS class under three failure scenarios. The healthy state is shown in Figure 2(a) and labeled as scenario 1. Figure 2(b) illustrates scenario 2 where flows 1 and 2 experience 0.1 Tbps and 0.2 Tbps of demand loss, respectively. Hence, the total loss for this QoS class adds up to 0.3 Tbps. In scenario 3, shown in Figure 2(c), only flow 1 loses 0.1 Tbps traffic, so the total loss is 0.1 Tbps. The demand loss for this QoS class is thus 0.3 Tbps — the highest loss across all scenarios. To obtain the availability for this QoS class, we first compute the availability of each flow. The demand of flow 1 is fully satisfied in only the no-failure case (scenario 1). As a result, its availability is computed as the probability (fraction of time) that the network is healthy: $1 - \frac{10\ hours}{1\ year + 10\ hours} - \frac{20\ hours}{3\ years + 20\ hours} = 99.81\%$. The demand of flow 2 is fully satisfied in scenarios 1 and 3, hence its availability is $1 - \frac{20\ hours}{3\ years + 20\ hours} = 99.92\%$. The availability of this QoS class is the lowest availability across both flows, which is 99.81%. To simplify the calculation of latency stretch in this example, we assume the latency of each link to be 1. Because the shortest tunnel latencies for both flows are 1, their bandwidth-weighted latency stretches in the healthy state (scenario 1) are both 1.5, as shown in Figure 2(a). In scenario 2 (Figure 2(b)) the latency stretch values for flows 1 and 2 are $\frac{0.2\times1+0.1\times3}{0.2+0.1} = 1.67$ and $\frac{0.1\times2+0.3\times2}{0.1+0.3} = 2$, respectively. Similarly, in scenario 3 (Figure 2(c)) the latency stretch values for flow 1 and 2 are $\frac{0.2\times2+0.1\times3}{0.2+0.1} = 2.333$ and $\frac{0.3\times1+0.3\times2}{0.3+0.3} = 1.5$, respectively. We then associate each failure scenario's probability to the corresponding latency stretch value to construct the latency stretch distributions.

3.3 Failure Modeling

High-fidelity failure modeling is important for network planning to meet SLOs. Hence, modeling failure scenarios is an essential component in calculating the risk metrics that we defined in the previous section. The goal of failure modeling is to estimate the likelihood of a failure scenario as well as the duration of the failure event. In this section, we explain Facebook's production failure model.

3.3.1 Characterizing Failure Events

We use two main variables to characterize failure events in our backbone:

(i) ***Time Between Failures*** *(TBF)* represents the duration between the recovery and the occurrence of two consecutive failures. This metric captures how reliable a network component (such as switch, linecard, or a fiber path) is. For most components in our backbone network, TBF tends to be thousands or even tens of thousands of hours.

(ii) ***Time To Repair*** *(TTR)* measures how long each failure event lasts. This metric depends on the efficiency of the network operation. Some failures (e.g., subsea fiber cuts) are more difficult to repair than others (e.g., switch failures).

Our experience indicates that fiber-related failures in our

backbone are the most devastating failure scenarios in terms of capacity loss and time to repair. As a result, our primary focus to model failures is on fiber-related issues.

Each fiber i under each failure scenario j is represented with a tuple $(TBF_{i,j}, TTR_{i,j})$. We use historical data analysis to estimate the values in each tuple. However, modeling every fiber in the backbone individually adds excessive complexity and will overwhelm the system. We need an intelligent clustering method to model fibers with similar features together. Moreover, we cannot completely rely on empirical observations. For instance, newly deployed fibers do not have historical failure data. As a result, we model $TBF_{i,j}$ and $TTR_{i,j}$ based on known features such as the length of the fiber and its supplier. The next section describes how we address these challenges.

3.3.2 Capturing Common Features and Data Sources

A naive approach to model failures is to use past failure events to compute TBF and TTR from historical data. However, there are practical challenges with this approach. First, rare failure events may not have enough historical data to faithfully compute their TBF and TTR. Second, data can be noisy. In particular, repair times are often recorded manually in our ticketing system, which may not be completely accurate. Third, data sources may belong to different administrative domains. For instance, leased fibers are operated by third-party vendors and we may not have access to the complete failure data. To address these challenges, we use a combination of common features and several data sources to model the failure characteristics of each fiber as accurately as possible.

Common Features. Each fiber is different, but there are common features that we can use to characterize a fiber without having its exact TBF and TTR. Below are the failure features we use in our system.

- *Fiber length:* Longer paths are more likely to experience fiber cuts due to greater surface area.
- *Vendor:* Fibers from certain vendors are more reliable than others, depending on their physical characteristics, operation quality, and contractual obligations to us.
- *Operational ownership:* Some fibers are purchased from the builder directly, while others may be leased or bought from indirect parties. We expect that without direct access to the fibers, subcontracted fibers have longer repair times.
- *Install type:* Subsea fibers are known to have longer repair time due to the difficulty in accessing the fiber or limited supply of maintenance ships. Similarly, aerial fibers are expected to have higher failure rates compared to buried fibers because the fiber is exposed to disasters and accidents.
- *Geographical region:* Failure rates can be higher in certain areas with frequent natural disasters, e.g., hurricanes. Repair times vary based on the weather condition, and can grow because of catastrophic events.

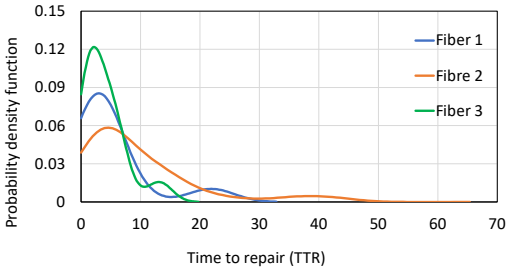


Figure 3: Distribution of time to repair for subsea fibers.

- *Urban density:* Fibers are more likely to be impacted in urban settings due to more frequent human activities, and hence accidental fiber cuts.
- *Shared Risk Link Group (SRLG):* Some fibers may fail together due to shared conduit or geographical proximity. An SRLG is considered as a single entity, hence, in our risk simulation system, we consider each SRLG as a single failure scenario.

Data sources. The impact of each feature can be analyzed using real-world data. We use three data sources for this purpose. (*i*) *Operational tickets:* Our Network Operation Center (NOC) maintains hundreds of incidents ticketed to our vendors. Each ticket contains confirmed failure information such as failed links, downtime, and failure root causes. We use this service as a historical benchmark for availability. However, the data is manually maintained, hence, the accuracy and coverage are both limited. (*ii*) *Continuous measurements:* We monitor counters from both IP-layer switches and optical-layer transponders and ROADMs. IP counters are collected every minute via the standard monitoring protocol SNMP. Optical counters are collected every three minutes using our optical-layer monitoring protocol TL1. To identify failures, we look for the Loss of Signal (LOS) on the Optical Service Channel (OSC) accompanied by the loss of IP links. (*iii*) *Fiber lifetime:* The above data sources both report discrete failure incidents, yet some fibers may not have failure events in recent years. Thus, we use the fiber lifetime dataset from our fiber inventory to compute the uptime of each fiber.

3.3.3 Failure Modeling Framework

We develop a failure modeling framework to best utilize the above data for estimating the failure model parameters. In particular, we take a two-step process.

1. Clustering. We start with a list of fibers and their failure characteristics from the data sources described in the previous section. Each record contains $< f_1, f_2, \ldots, f_k, TBF_t, TTR_t, TBF_d, TTR_d >$, where f_k is the k^{th} element in the feature set, TBF_t and TTR_t are the TBF and TTR values from the tickets, and TBF_d and TTR_d are those from the continuous measurements. We then use a Bayesian clustering algorithm to identify groups of fibers that share

	MTTR	MTBF
Subsea fibers vs. non-subsea fibers	90×	36×
Leased fibers vs. non-leased fibers	1×	0.4×
Fibers in the most different region vs. fibers in other regions	2×	5×

Table 2: MTTR and MTBF of different fiber categories.

similar failure characteristics. The output of this step is a set of clusters $(C_1, \ldots, C_g)$, where each C_i contains a set of fibers.

2. Bayesian Hierarchical Model. Next, for each cluster, we use an exponential hierarchical model to fit the distribution of TBF and TTR separately. We find the mean TBF (MTBF) and mean TTR (MTTR) from both fitted curves and use them in RSS (§3.4). The accuracy of this model is evaluated in §4.1.

Operational observations. In the following, we summarize some of our empirical measurement results that have provided inspirations for our failure modeling.

Subsea TTR follows arbitrary distribution. Figure 3 shows the TTR distributions of three subsea fibers from our empirical failure data source. Each subsea fiber has a unique TTR distribution, due to its physical properties such as the length and placement under the ocean that determines the accessibility for repair endeavors. This observation deviates from a common technique to use a simple exponential distribution for TTR in two major ways. First, unlike exponential distribution, there is a lower bound for the TTR. This lower bound corresponds to the physical time constrains such as the time to secure permits to enter the water and the sailing time. Second, the distribution is multi-modal since each subsea fiber has distinct parts with different failure profiles depending on the depth under water.

Impact categories. We categorize three key factors that have significant impacts on the failure model: whether a fiber is subsea, leased, or belongs to a particular region. Table 2 shows the relative impact of different fiber categories on MTTR and MTBF. If a fiber is subsea, its MTTR is 90× longer than that of non-subsea fibers, and its MTBF is 36× longer. This is because subsea fibers are less frequently cut, but once they are cut, they will take much longer to be repaired. Leased fibers have similar MTTR as non-leased (Facebook-owned) fibers, but are 2.5× more likely to fail in terms of MTBF. For the region factor, we select the region with the largest difference from the rest ones, and we observe a 5× difference in MTBF and 2× difference in MTTR. These results show the drastic differences between fiber types and indicate the importance of clustering fibers into appropriate failure groups.

3.4 Risk Simulation System (RSS)

3.4.1 System Design

RSS performs periodic simulations (e.g., every 30 minutes) to report the risk metrics defined in Section 3.2. Figure 4

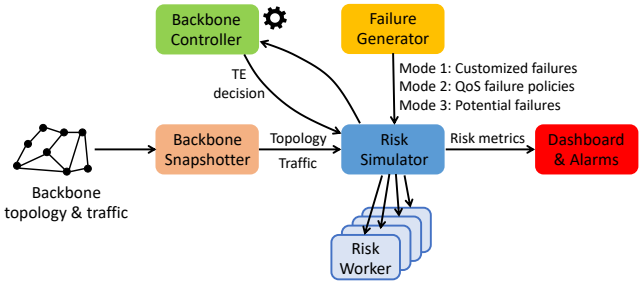


Figure 4: Risk Simulation System architecture.

depicts our risk simulation pipeline. For each simulation run, the *Backbone Snapshotter* polls the backbone routers for the latest IP topology and traffic demand. The *Failure Generator* generates hypothetical failure scenarios to be simulated. The *Risk Simulator* takes in such information to simulate routing on the residual topology under different failures. To simplify system implementation, we reuse the binary of our centralized *Backbone Controller* which computes the TE solution using a global optimization formulation. Since routing simulations for different failure scenarios are independent, we shard them onto a number of *Risk Workers* for parallelization. The risk metrics are calculated from the worker instances and displayed on a real-time risk dashboard. Risk values higher than pre-defined thresholds will raise production alarms.

Calculation of the risk metrics requires failure probabilities, which can be derived from MTBFs and MTTRs — the mean values of the failures' TBF and TTR distributions in Section 3.3. For most failures, the TBF and TTR follow exponential distributions. However, the TTR for subsea fibers is arbitrary and hard to model. As a result, we estimate their metrics from our empirical failure observation in production.

Suppose a failure scenario includes n failed fibers $\{f_1, f_2, ..., f_n\}$. For a particular fiber f_i, the probability of being available is $A(f_i) = \frac{MTBF_i}{MTBF_i + MTTR_i}$, and the probability of being under failure is $P(f_i) = 1 - A(f_i)$. Therefore, the probability of the entire failure scenario is $P(f_1, f_2, ..., f_n) = \Pi_{i=1}^{n} P(f_i)$, and the available probability is $A(f_1, f_2, ..., f_n) = 1 - \Pi_{i=1}^{n} P(f_i)$. Given the failure probability of each failure scenario, the risk metrics can be calculated using Algorithm 1.

RSS supports three modes of operation. Mode 1 is for fine-grained simulation of customized failures that are expected to happen in the near future. For example, it is part of our decommission workflow where capacity is removed, or migrated from one fiber to another, according to the backbone expansion plan. Before decommission tasks are carried out, RSS is used to generate failure scenarios that reflect the decommission plan. The risks associated with these scenarios are then taken into account to ensure there is sufficient protection capacity in the network. Mode 1 also serves for risk monitoring and mitigation under natural disasters. For instance, in response to a hurricane forecast, we simulate failure scenarios relating to the hit regions and shift traffic as neces-

sary. Similarly, as COVID-19 goes on, we plan to simulate failure scenarios for specific geological locations based on the severity of the pandemic.

Mode 2 is fine-grained simulation of pre-defined failure scenarios for different QoS classes given their protection policies. In production, the protection policies include four categories of critical fiber failures: (1) single fiber failures, (2) SPOFs where multiple SRLGs use fibers in the same conduit or have the same geographical proximity, (3) dual subsea failures where two subsea fiber paths fail simultaneously, and (4) dual DC failures where two fiber paths from the same DC fail simultaneously. These four categories include over 6000 failure scenarios in the Facebook backbone. As described in Section 3.1, different QoS classes protect against different failure categories. QoS classes 1 and 2 carry our critical services and have full protection against all the above failure categories. QoS class 3 (default class for our internal traffic) relaxes on dual DC failures, because they account for over 50% of the failure scenarios but are less likely than single fiber failures and SPOFs and have less severe consequences compared to dual subsea failures. QoS class 4 (background bulk data transfers) is best-effort service without failure protection. We use Mode 2 in RSS to validate the QoS performance and guide network maintenance.

Mode 3 is coarse-grained simulation of a large number of potential failures in the backbone, where the exact number is determined by a cutoff threshold. The cutoff threshold can be defined in different forms, such as by failure probability, the number of concurrent failures, MTTR, or the protection cost (in terms of the protection capacity, construction cost, and maintenance work), under the intuition that we value failures that are more likely to happen, take a longer time to repair, or are affordable to protect against. We typically have a quick scan of the network health considering *millions* of failure scenarios. This simulation mode must be coarse-grained given the large number of failure scenarios. We bypass the computation-heavy global TE optimization with efficient routing approximations, which will be discussed in Section 3.4.2. This mode of operation offers a tradeoff between simulation accuracy and runtime, and the choice depends on the number of failure scenarios and how close to production the simulation needs to be (e.g., replaying production situations in Mode 1 and 2 vs. a big picture of the network in Mode 3).

3.4.2 System Optimizations

RSS is implemented using around 18,000 lines of C++ code. This system is highly optimized for fast execution time. Today, it can finish a fine-grained simulation of one failure scenario in an average of 250 seconds and a coarse-grained simulation of a failure scenario in 0.1 second. Important performance improvements attribute to the following optimizations.

Parallelization. Our risk simulation is highly parallelizable by nature. Our first implementation was based on a two-layer master-slave architecture where the failure scenarios were distributed across the slave nodes and the simulation results were aggregated to the master. The master node was overwhelmed with the aggregation load when we scaled to 50 slaves, hence we added another layer in the middle to aggregate the intermediate results generated by slaves and then transmit them to the master. Today, we use tens of aggregators and hundreds of slave nodes to optimize the execution time of RSS.

Routing simplification. Our fine-grained simulation emulates the production backbone by executing the TE algorithm when a failure happens. This process is computationally expensive, especially when we simulate a large number of failure scenarios. Thus, for coarse-grained simulations, we simplify the TE implementation with shortest-path routing of small units of sub-flows. Specifically, we split each traffic flow in the backbone demand matrix, usually hundreds of Gbps or several Tbps big, into minimal sub-flows around 1 Gbps and pack them one by one onto the shortest path until there is no remaining capacity in the network. The result is close to production TE when the sub-flows are sufficiently small.

Merge duplicate failures. Different fiber failures can result in the same failure scenarios on the IP layer, which can be effectively merged during risk simulation. For example, failures of different SRLGs may cause different fiber spans to be down, but they create the same failure scenario for the IP links over the fiber paths traversing any of these fiber spans. Because the risk is ultimately simulated on the IP-layer network, RSS translates the fiber failures into IP-layer failures and merges duplicate failure scenarios. The failure probability of a merged failure scenario equals to the sum of the probabilities of each individual failure event.

Identify dominating failures. We further reduce the simulation time by only simulating failures with severe consequences. We define dominating failures as the ones that contain subsets of other failures. For example, the failure scenario with fiber cuts $\{f_1, f_2\}$ is a dominating failure of single fiber failures f_1 and f_2 alone. Note that this simplification only applies to the calculation of demand loss, which does not rely on the probabilities of failure scenarios (the other two risk metrics, latency stretch and availability, need to factor in failure probability). This is because the failure probability of a dominating failure is much smaller than the probabilities of its subset failures. In production, we usually use this approach for the Mode 1 simulation, where the demand loss of expected failure events is a critical signal for network maintenance.

4 Evaluation

In our daily operation, we keep monitoring the health of the backbone network with the risk metrics (Section 3.2) produced by RSS (Section 3.4). Here we report the risk measurement results from November 2019 to September 2020

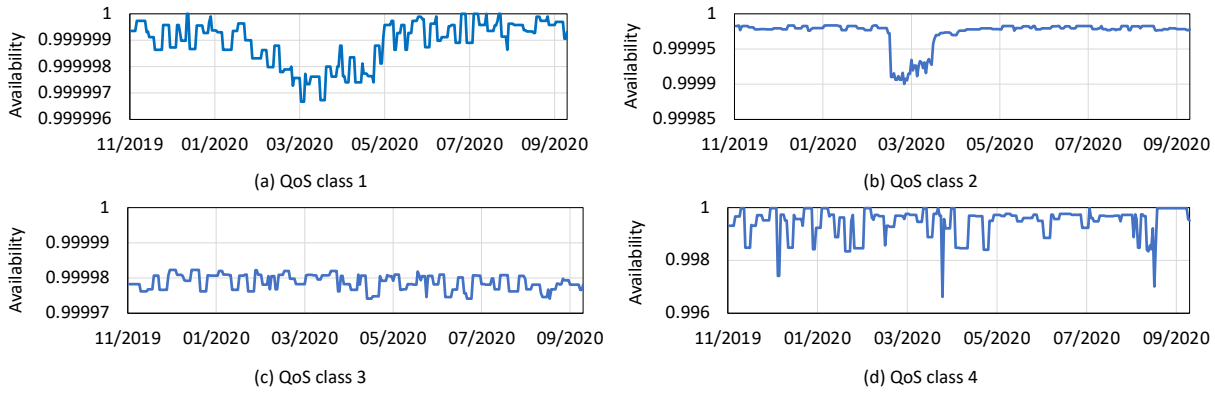


Figure 5: Availability over time for each QoS class.

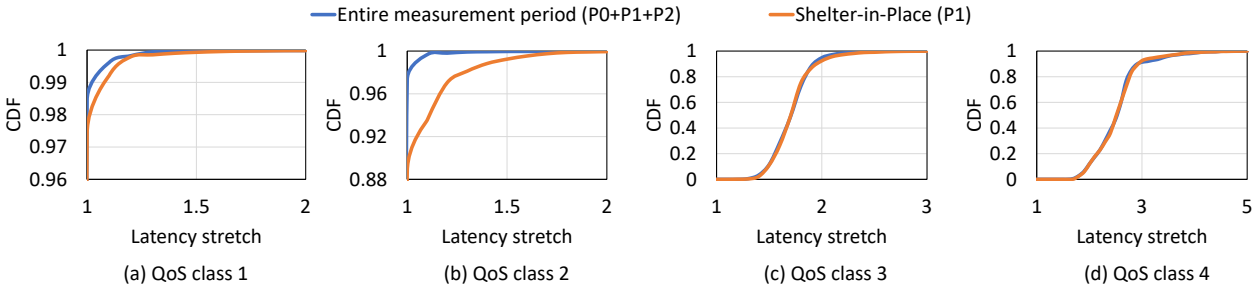


Figure 6: CDF of per-flow latency stretch for each QoS class.

and share our operational experience to survive the extreme conditions under COVID-19.

4.1 Observations with RSS

An important input into RSS is production traffic. During COVID-19, we observed an increase in network risk triggered by significant traffic surges (Figure 1). However, most risk metrics remained at a reasonable level, indicating that our backbone was robust under the COVID-19 stress test.

High availability. As shown in Figure 5, different QoS classes all achieved high availability over time, constantly reaching the SLO goals. The traffic increase during social distancing mostly related to the user traffic in QoS class 2, causing its availability to drop sharply from around 0.99998 to 0.9999. QoS class 1 also experienced a minor availability reduction, as the traffic for highly critical user services increased as well. The change was smoother compared to QoS class 2, because QoS class 1 also contains system control traffic irrelevant to user behaviors. QoS classes 3 and 4 that mostly comprise machine-to-machine computational traffic showed no obvious decrease in availability. These results suggest that availability is highly sensitive to traffic volume. On the positive side, our backbone infrastructure is over-provisioned, making it robust against the unprecedented traffic surge.

Low latency stretch. From Figure 6, we see a minimal change of latency stretch during the shelter-in-place period (P_1). Recall from Section 3.1 that QoS classes 1 and 2 use CSPF routing. As shown in the figure, over 97% of the flows in QoS classes 1 and 2 had a latency stretch of 1 throughout the entire measurement period ($P_0 + P_1 + P_2$), meaning they went through the shortest paths. The latency stretch of QoS class 2 was slightly higher than that of class 1, because the bandwidth for QoS class 2 is allocated after class-1 flows are fully accommodated. Similar to the availability results, latency stretch degraded the most in QoS class 2 during the shelter-in-place period (P_1) due to the traffic increase. Yet, the stretch still remained low regardless of the COVID-19 increase: it stops at 1.7 for most flows in QoS class 2, and at 1.4 for QoS class 1, though with a long tail not shown in the figures. QoS classes 3 and 4 use a combination of KSP and MCF routing, so they generally take longer paths. The mean latency stretch for QoS class 3 was 1.71, and 2.53 for QoS class 4. COVID-19 caused little increase of latency stretch for these two traffic classes, which is consistent with the trend of traffic growth and availability change.

Accurate failure modeling. We evaluate the accuracy of our failure model (§3.3) by comparing the TTRs and TBFs of observed fiber failures in North America against our model's predictions. As shown in Figure 7, our failure model is close to the actual observed values. To quantify the difference between prediction and observation, we perform a Kolmogorov-Smirnov (KS) test [1] on the null hypothesis that the measurements and the model-generated samples are drawn from the same distribution. We report the KS statistic and p-value in Table 3. Both p-values are large, meaning the two distributions match. Lastly, we directly compute the prediction accuracy, as the difference between each observed and predicted value,

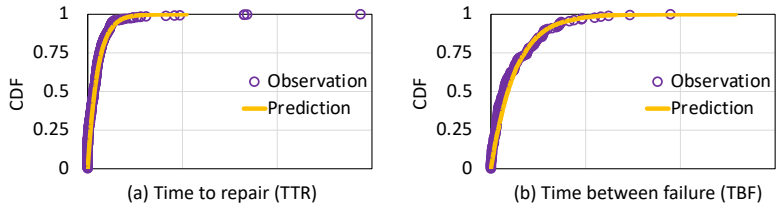


Figure 7: CDF of TTR and TBF from failure prediction vs. observation (the values on the x-axis are anonymized).

	KS stats	$p-$value	accuracy
TTR	0.05	0.25	94%
TBF	0.03	0.47	98%

Table 3: Kolmogorov-Smirnov (KS) test statistics and accuracy of TTR and TBF models.

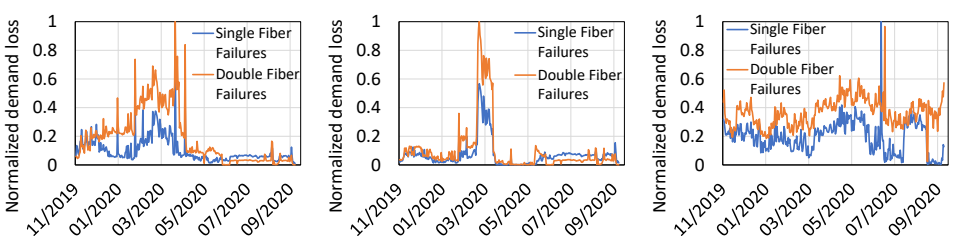


Figure 8: Normalized demand loss per QoS class.

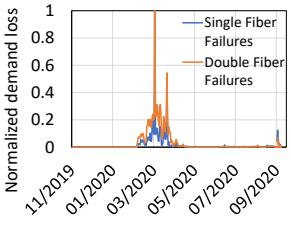


Figure 9: QoS class 2 downplayed into class 3&4.

divided by the observed value. Our model achieves 94% accuracy for TTR and 98% accuracy for TBF prediction.

4.2 Risk Mitigation

Demand loss as a guide. Demand loss is a rigid risk metric, which captures the highest traffic loss across all simulated failure scenarios. It guides our operations for mitigating potential risk. Figure 8 shows the demand loss for each QoS class over time. For confidentiality reasons, we normalize the numbers in each QoS class against the highest loss value. We group different types of failures that we track into two categories: single fiber failures and dual fiber failures. These categories capture the major failures we protect against in production. The figure shows that the demand loss increased during the P_1 phase for QoS classes 1 and 2. In particular, the mean value of risk during the shelter-in-place phase (P_1) increased by 80% compared to the pre-COVID period (P_0) in QoS class 1, and by 3.6× in QoS class 2. QoS classes 3 and 4 did not have a significant change in their demand loss values during the pandemic, and the loss increase in March 2020 was due to traffic migration between regions because of an internal policy change. Note that dual failures, though less common in practice, induce 2.14× higher loss on the fabric than single failures, on average. This worst-case analysis makes us operate the network conservatively, which was especially beneficial during the pandemic period.

QoS downplay. Another key technique to save capacity for the most critical traffic is to adjust the QoS assignment. By default, all traffic from a service is assigned the same QoS class. However, a service usually contains traffic from both user requests and system metadata, whose importance should be differentiated. This coarse-grained performance

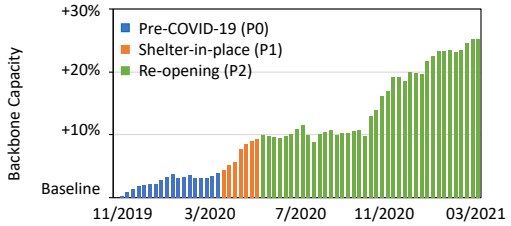
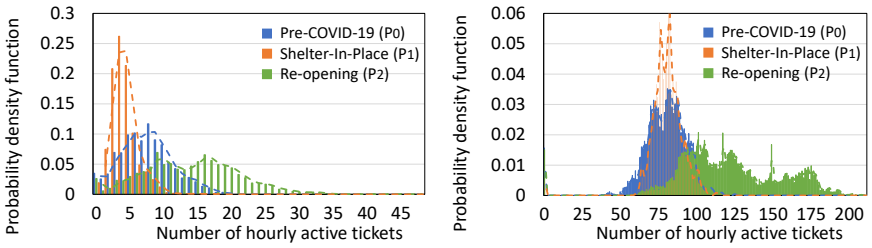


Figure 10: Backbone capacity measured per week.

isolation causes over-protection of unimportant traffic, and the resulting capacity waste should be recycled for traffic increase. We have developed an internal system that leverages inference mechanisms to identify the true traffic priorities and correct their QoS labeling. For example, we find that on-off and periodic traffic patterns are common signals for machine-created traffic, which can be downgraded to a lower class. Figure 9 zooms into QoS classes 3 and 4 in Figure 8(c) and shows the demand loss of the downplayed traffic. The loss only appears during the shelter-in-place phase (P_1) as the result of traffic shift to alleviate the stress from QoS class 2.

Proactive capacity enhancement. We deploy optical wavelengths periodically to augment the capacity of our backbone. Figure 10 shows the weekly measurement of backbone capacity over a year, normalized by the capacity value before P_0. We observe an aggressive capacity increase starting from the P_1 phase compared to the pre-COVID times (P_0). This capacity increase is also visible in Figure 8, leading to a significant drop in demand loss in April and May 2020. We observe a continued capacity increase during the re-opening phase (P_2). Although social distancing during the COVID-19 pandemic paused most of our site work for deploying new



(a) Optical layer

(b) IP layer

Figure 11: Impact of COVID-19 on hourly active failure tickets.

Failure category	Mean # of tickets			$p-$value for
	P_0	P_1	P_2	P_0 and P_1
IP	80.56	80.28	121.4	0.44
Optical	7.67	3.72	14.0	$\ll 0.001$

Table 4: Statistical comparison on the number of active tickets during our measurement phases.

fibers, we had sufficient dark and under-provisioned fibers from previous planning cycles. Our "plan ahead" strategy gave us enough headroom for emergency capacity enhancement, and our fully-automated optical management system allowed us to provision wavelengths remotely.

5 Insights on RSS Improvement

Besides a stress test of our network infrastructure, COVID-19 and the resulting social distancing also created unusual situations beyond normal assumptions of network operations. These edge cases have given the networking community a unique opportunity to rethink fundamental design assumptions of networks. In this section, we share our recent progress on failure modeling and traffic forecasting to shed light on future evolution of risk simulations.

5.1 Responsive Failure Modeling

Our failure modeling (Section 3.3) uses years of failure measurement data to ensure model accuracy and stability, and it is accurate in the long run (Figure 7). However, prior studies claimed network failures are often caused by human activities and network operations [19, 20, 29, 30, 38]. Indeed, we confirm that the lack of human activities during the shelter-in-place phase (P_1) as well as the increase of network upgrade and capacity augmentation activities during the re-opening phase (P_2) changed the failure characteristics. In response to the change, we recalibrated our failure model to increase the weight of failure statistics during the P_0, P_1 and P_2 periods.

The recalibration relies on Facebook's centralized failure ticketing system, which automatically detects network failures and infers possible root causes. For this purpose, we categorize failure tickets into two groups: (*i*) optical-layer failures (e.g. fiber/amplifier/transponder issues) and (*ii*) IP-layer failures (e.g. router/interface down). For each group, we record the number of active failure tickets every hour and plot the probability density function of each phase in Figure 11. Our observations are as follows.

Optical-layer failures. Figure 11(a) compares the probability density function of hourly active optical-layer failure ticket numbers among the pre-COVID (P_0), shelter-in-place (P_1) and re-opening (P_2) phases. The dashed lines in the figure represent the moving average of each phase. We observe reduced optical-layer failures during P_1 compared to P_0. We attribute this finding to the significant reduction in human activities at our backbone facilities during global social distancing. For instance, limited construction work can lead to fewer fiber cuts, and fewer human contractors on-site may result in fewer accidental link flaps, as suggested in prior work [19]. We observe more active optical failure tickets in the re-opening phase (P_2), as our network operation team was actively augmenting the capacity of our backbone.

IP-layer failures. Figure 11(b) shows the probability density function of hourly active IP-layer ticket numbers during the P_0, P_1 and P_2 phases. In contrast to optical-layer failures, we find no significant changes in IP-layer failures between P_0 and P_1, and the two distributions largely overlap with each other. This is likely because IP-layer tickets, such as router/interface hardware failures, are mostly mechanical issues and do not correlate with human activity. However, similar to the optical-layer failures, in the re-opening (P_2) phase, the IP layer also experienced more active failure tickets due to our aggressive capacity provisioning operations. Moreover, since the pandemic continued to enforce limitations on our failure repair staff, the P_2 phase suffered from longer repair times.

Confirmation with statistical hypothesis test. The results in Figure 11 suggest that optical and IP-layer failure distributions behave differently during the P_1 phase. To confirm this observation, we apply a statistical hypothesis test on the time-series distribution of active number of tickets between the P_0 and P_1 timelines and set the null hypothesis to be: *the means of the distributions are the same*. We apply Welch's t-test on the optical and IP-layer categories separately to validate the null hypothesis. Table 4 reports the mean number of active tickets and the corresponding $p-$values for each category. Considering a $p-$value threshold of 0.01, the null hypothesis cannot be rejected for IP-layer tickets, suggesting the pandemic did not have a significant impact on the IP-layer failures during P_0 and P_1. In contrast, the null hypothesis is rejected for optical tickets suggesting that the average number of active optical tickets changed in a statistically significant manner. Note that we do not run hypothesis tests over P_2 because its mean values in both the IP and optical layers differ a lot from those of P_0 and P_1, which already indicates

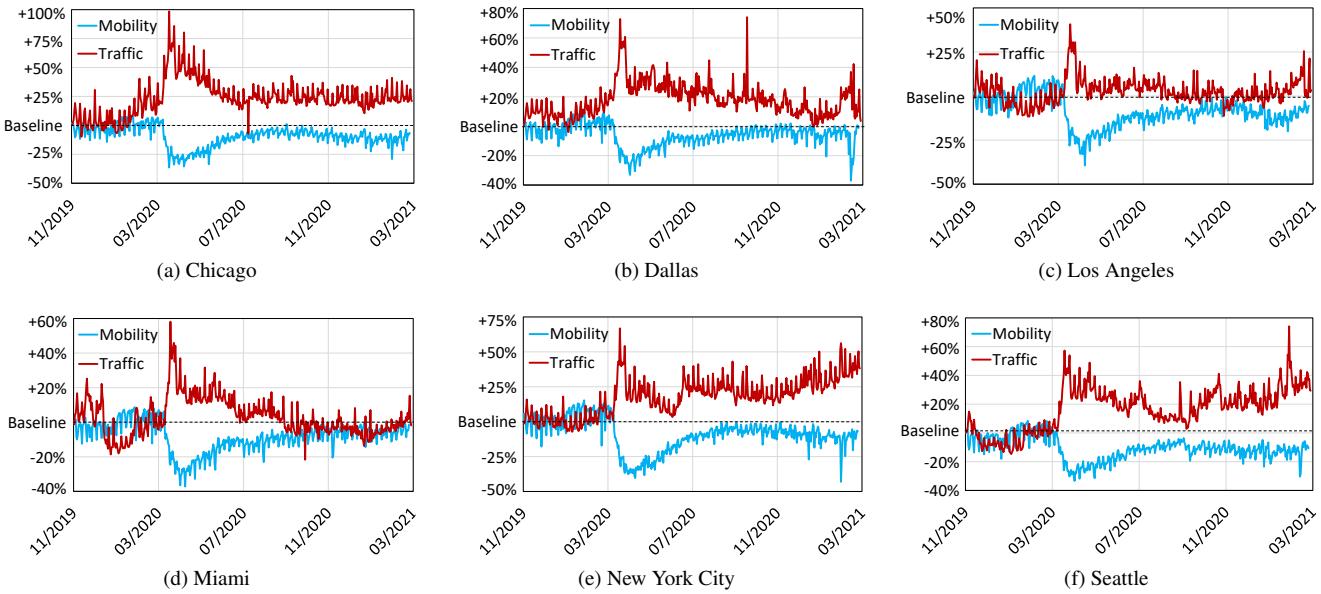


Figure 12: Traffic volume and mobility patterns in six US cities during the COVID-19 pandemic.

differences in the probability distributions.

Insights. These findings call for responsive failure modeling. When special events cause failure characteristics to change, the failure prediction model should be adjusted to rely more on recent failure measurement points. However, the model stability might be at stake with short-term data collection, and the challenge lies in balancing stability and agility to have an accurate model. The COVID-19 crisis required us to respond quickly, and the fast development of the pandemic gave us little time for drastic redesigns of the failure model. The various failure generation modes in RSS make it adaptive to different failure models. For instance, our customized failures in Mode 1 are designed for failure scenarios of particular interests that may deviate from the failure model. We leveraged this feature to feed RSS with short-term failure statistics for close monitoring of the network health during COVID-19. The failure model is hard to change in Modes 2 and 3, hence, we applied a scaling factor to the failure distributions to generate more failures. These false positives helped us operate the network more cautiously during the pandemic. Moving forward, we are working on more responsive failure modeling with a sliding window that automatically assigns weights to different measurement periods.

5.2 Traffic Prediction with External Signals

The risk observations we report throughout this paper use current production traffic as input, yet RSS can also take in projected traffic to forecast future risk. At Facebook, we perform demand forecasting every quarter to predict the traffic volume in the next 6 months to one year. Our prediction used to be accurate, but the traffic grew beyond the predicted upper bound since the pandemic started. At the peak, we saw a 26% difference between actual and predicted upper-bound traffic.

In our operational experience, we have seen rich examples of how external non-networking signals can be leveraged to aid network management. For instance, it is common practice to strengthen the guard on PoP or DC regions that have received hurricane warnings, and we keep a close watch on traffic blackholing in areas with frequent armed conflicts. In this section we demonstrate that human *mobility* metrics can also be used to better predict the traffic volume.

To demonstrate this finding, we use population mobility data from the SafeGraph [5,41] database built from 20 million mobile devices. As an approximation of mobility, we sum the total number of trips that take place in a geographical region based on aggregated cellphone GPS data, and normalize it by the population of the trip origin. We consider six major US cities and plot the variations of traffic and mobility over time in Figure 12. Interesting findings imply opportunities and challenges in our proposal of mobility-aided traffic prediction.

Negative correlation between traffic and mobility. Figure 12 shows the traffic volume and mobility rate normalized to their corresponding averages during the P_0 phase. While there is a fair amount of overlap between traffic and mobility in P_0 across the cities, we observe a strong negative correlation between traffic and mobility since the start of P_1. We see the general trend that when mobility drops, traffic increases; and as mobility increases slowly, traffic falls as well. The sporadic spikes of mobility and traffic also match well, forming zigzags in opposite directions. The gap between the traffic and mobility curves closes down in the P_2 phase when the

cities started to re-open and social distancing reduced.

Variations across cities. Each city shows some uniqueness despite the same trend. Chicago, Dallas, Los Angeles, and Miami have similar patterns that network traffic gradually decreased while mobility continued to increase after the pandemic peak in mid-Mach 2020. In Chicago, traffic increased by 99% and mobility rate decreased by 36% during the shelter-in-place phase (P_1) compared to the P_0 phase baseline. Dallas, Los Angeles, and Miami had around a 40-70% traffic increase and a 35% mobility drop. Roughly, the drop in mobility rate corresponds to different levels of traffic reduction across cities. For example, in Miami and Los Angeles the traffic volume almost returned back to normal in the P_2 phase, while Chicago still showed a 25% average traffic increase, and Dallas had around 10% average traffic increase. On the other hand, New York City and Seattle have contrasting patterns, with both an uptake in traffic and a downtake in mobility appearing since November 2020. For Seattle, we also observe ups and downs in traffic volume, with occasional spikes up to 74%.

Insights. An intrinsic limitation of traditional network management is the complete reliance on in-network signals. It fails to track social influences on the network infrastructure, which has been proven to be heavily underestimated during COVID-19. Our mobility case study shows the potential benefit of embracing offline signals from the outside world. On occurrences of social events, we can make mobility the main signal for traffic prediction. As Figure 12 shows, traffic peaks can be inferred from the steep drops in mobility rate. However, it is challenging to estimate the traffic volume when it recovers, as mobility rate does not show significant changes in that case. We may need other signals to understand user behaviors better. Fortunately, though, risk management cares about worst-case scenarios. Thus, an accurate prediction of the peak traffic is already a big win for risk prevention.

6 Related Work

Risk-aware network management. There have been recent proposals to apply risk to capacity planning [4, 6] and traffic engineering [8, 12, 31, 46]. The definition of risk is different across proposals, including probabilistic models of failures [6, 8], revenue shortfall [31], early signs of failures (e.g., hardware abnormality and performance degradation) [46], user-specified undesirable events and their associated probabilities [12], and the likelihood of losing customer traffic during planned network changes [4]. In this paper, we describe Facebook's definition of risk as a set of SLO-related metrics that quantify the impacts of potential failures. We are also the first to apply risk simulation to backbone management and to develop a production system.

Internet under COVID-19. There are reports on the impacts of COVID-19 on the Internet [10, 13, 17, 24, 32, 33], as well as measurement studies on the PoP traffic [7], mobile traffic [28], Internet traffic [15], and cybercrime [47] during the pandemic. We share similar observations on the traffic increase, but we present a comprehensive study on the impact of COVID-19 from the perspective of risk management.

Backbone failures. Prior work on understanding backbone failures includes statistical modeling [3, 9, 42] and real-world measurements on both the optical layer [18, 19, 38, 43] and the IP layer [11, 14, 23, 25, 26, 29, 30, 35]. Our failure analysis confirms the observation in previous papers that a good proportion of failures are human-related [19, 20, 29, 30, 38].

Traffic classification with QoS. QoS can provide differentiated performance for different categories of traffic [16, 48]. There have been rich discussions on traffic classification methods in the prior work [36, 39, 40, 44]. In recent years, with the development of SDN, traffic classification can be deployed with centralized control on a private enterprise network [34]. Facebook follows these discussions and categorizes the backbone traffic into four classes of QoS. To maximize user satisfaction while considering network risks, our traffic classification scheme can dynamically adjust traffic flows' QoS categories so as to prioritize critical traffic flows and guarantee service level objectives.

7 Conclusion

This paper introduces RSS, a risk simulation system deployed at Facebook. We present our risk analysis with RSS during the COVID-19 pandemic period and beyond. Motivated by the surge of traffic volume, we define risk metrics to quantify the impact of COVID-19 and show our strategies to mitigate the risk. We keep the network running at low risk levels during this challenging time and propose that having responsive failure modeling and using external signals such as human mobility can help understand the social impacts on the network to further improve network management. Our experience and insights are useful for managing large-scale backbones in the post-pandemic world, where we are likely to face an ever-growing demand for online services. We hope that our experience can inspire and guide practitioners towards embracing risk-driven network management and ultimately making it a key strategy for ensuring high availability of networked services.

Acknowledgments. Many people in the Network Infrastructure team at Facebook have contributed to RSS over the years. In particular, we would like to acknowledge Max Noormohammadpour, Siyang Xie, Bob Kamma, Abhinav Triguna for their early work on RSS. We also would like to thank Arash Vakili, Ariyani Copley, Brian Bierig, Debottym Mukherjee, Lu Huang, Tyler Price, Sandrine Pasqualini for their operational support during the COVID-19 time period. We would like to acknowledge Facebook for the resource it provided for us. And finally, we thank Omar Baldonado, Gaya Nagarajan, our shepherd Haonan Lu, as well as our anonymous reviewers for their comments. This work was partly supported by NSF grants CNS-2008624 and ASCENT-2023468, as well as by SystemsThatLearn@CSAIL Ignite Grant.

References

[1] Kolmogorov-Smirnov test. http://www.mit.edu/~6.s085/notes/lecture5.pdf.

[2] State Shelter-in-Place and Stay-at-Home Orders, 6 2020. https://www.finra.org/rules-guidance/key-topics/covid-19/shelter-in-place.

[3] Anuj Agrawal, Vimal Bhatia, and Shashi Prakash. Network and Risk Modeling for Disaster Survivability Analysis of Backbone Optical Communication Networks. *Journal of Lightwave Technology*, 37(10):2352–2362, 2019.

[4] Omid Alipourfard, Jiaqi Gao, Jeremie Koenig, Chris Harshaw, Amin Vahdat, and Minlan Yu. Risk based Planning of Network Changes in Evolving Data Centers. In *Proceedings of the 27th ACM Symposium on Operating Systems Principles*, pages 414–429, 2019.

[5] Nick Altieri, Rebecca L. Barter, James Duncan, Raaz Dwivedi, Karl Kumbier, Xiao Li, Robert Netzorg, Briton Park, Chandan Singh, Yan Shuo Tan, Tiffany Tang, Yu Wang, and Bin Yu. Curating a COVID-19 data repository and forecasting county-level death counts in the United States, 2020. https://arxiv.org/abs/2005.07882.

[6] Ajay Kumar Bangla, Alireza Ghaffarkhah, Ben Preskill, Bikash Koley, Christopher Albrecht, Emilie Danna, Joe Jiang, and Xiaoxue Zhao. Capacity planning for the Google Backbone network. In *ISMP (2015)*, 2015.

[7] Timm Boettger, Ghida Ibrahim, and Ben Vallis. How the Internet reacted to Covid-19. In *Proceedings of the 2020 ACM SIGCOMM conference on Internet measurement conference*, 2020.

[8] Jeremy Bogle, Nikhil Bhatia, Manya Ghobadi, Ishai Menache, Nikolaj Bjørner, Asaf Valadarsky, and Michael Schapira. Teavar: Striking the right utilization-availability balance in WAN traffic engineering. In *Proceedings of the ACM Special Interest Group on Data Communication*, SIGCOMM 19, page 29?43, New York, NY, USA, 2019. Association for Computing Machinery.

[9] Graham Booker, Alexander Sprintson, E Zechman, C Singh, and S Guikema. Efficient traffic loss evaluation for transport backbone networks. *Computer Networks*, 54(10):1683–1691, 2010.

[10] Joseph Brookes. Akamai data shows 30 percent surge in Internet traffic, April 2020. https://which--50-com.cdn.ampproject.org/c/s/which-50.com/an-extraordinary-period-in-internet-history-akamai-data-shows-30-per-cent-surge-in-internet-traffic/.

[11] Alberto Dainotti, Claudio Squarcella, Emile Aben, Kimberly C Claffy, Marco Chiesa, Michele Russo, and Antonio Pescapé. Analysis of country-wide Internet outages caused by censorship. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 1–18, 2011.

[12] Ferhat Dikbiyik, Massimo Tornatore, and Biswanath Mukherjee. Minimizing the risk from disaster failures in Optical Backbone Networks. *Journal of Lightwave Technology*, 32(18):3175–3183, 2014.

[13] Fastly. How COVID-19 is affecting Internet Performance, 2020. [Online; accessed 20-May-2020].

[14] Nick Feamster, David G Andersen, Hari Balakrishnan, and M Frans Kaashoek. Measuring the effects of Internet path faults on reactive routing. *ACM SIGMETRICS Performance Evaluation Review*, 31(1):126–137, 2003.

[15] Anja Feldmann, Oliver Gasser, Franziska Lichtblau, Enric Pujol, Ingmar Poese, Christoph Dietzel, Daniel Wagner, Matthias Wichtlhuber, Juan Tapiador, Narseo Vallina-Rodriguez, Oliver Hohlfeld, and Georgios Smaragdakis. The Lockdown Effect: Implications of the COVID-19 Pandemic on Internet Traffic. In *Proceedings of the 2020 ACM SIGCOMM conference on Internet measurement conference*, 2020.

[16] Victor Firoiu, J-Y Le Boudec, Don Towsley, and Zhi-Li Zhang. Theories and models for Internet Quality of Service. *Proceedings of the IEEE*, 90(9):1565–1591, 2002.

[17] Forbes News. Apple Data Shows Shelter-In-Place Is Ending, Whether Governments Want It To Or Not, 2020. [Online; accessed 20-May-2020].

[18] Monia Ghobadi, Jamie Gaudette, Ratul Mahajan, Amar Phanishayee, Buddy Klinkers, and Daniel Kilper. Evaluation of Elastic Modulation Gains in Microsoft's Optical Backbone in North America. In *Optical Fiber Communication Conference*, page M2J.2. Optical Society of America, 2016.

[19] Monia Ghobadi and Ratul Mahajan. Optical Layer Failures in a Large Backbone. *IMC*, 2016.

[20] Ramesh Govindan, Ina Minei, Mahesh Kallahalla, Bikash Koley, and Amin Vahdat. Evolve or Die: High-Availability Design Principles Drawn from Google's Network Infrastructure. In *SIGCOMM*, 2016.

[21] Chi-Yao Hong, Subhasree Mandal, Mohammad Al-Fares, Min Zhu, Richard Alimi, Chandan Bhagat,

Sourabh Jain, Jay Kaimal, Shiyu Liang, Kirill Mendelev, et al. 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*, pages 74–87, 2018.

[22] Mikel Jimenez and Henry Kwok. Building Express Backbone: Facebook's New Long-haul Network. *Facebook Engineering*, 2017. https://engineering.fb.com/2017/05/01/data-center-engineering/building-express-backbone-facebook-s-new-long-haul-network/.

[23] Ethan Katz-Bassett, Colin Scott, David R Choffnes, Ítalo Cunha, Vytautas Valancius, Nick Feamster, Harsha V Madhyastha, Thomas Anderson, and Arvind Krishnamurthy. Lifeguard: Practical repair of persistent route failures. *ACM SIGCOMM Computer Communication Review*, 42(4):395–406, 2012.

[24] Ella Koeze and Nathaniel Popper. The Virus Changed the Way We Internet, April 2020. https://www.nytimes.com/interactive/2020/04/07/technology/coronavirus-internet-use.html.

[25] Craig Labovitz, Abha Ahuja, and Farnam Jahanian. Experimental study of Internet Stability and Backbone Failures. In *Digest of Papers. Twenty-Ninth Annual International Symposium on Fault-Tolerant Computing (Cat. No. 99CB36352)*, pages 278–285. IEEE, 1999.

[26] Craig Labovitz, Roger Wattenhofer, Srinivasan Venkatachary, and Abha Ahuja. Resilience characteristics of the internet Backbone routing infrastructure. In *Proceedings of the Third Information Survivability Workshop, Boston, MA*, 2000.

[27] Hongqiang Harry Liu, Srikanth Kandula, Ratul Mahajan, Ming Zhang, and David Gelernter. Traffic engineering with forward fault correction. In *ACM SIGCOMM (2014)*, 2014.

[28] Andra Lutu, Diego Perino, Marcelo Bagnulo, Enrique Frias-Martinez, and Javad Khangosstar. A Characterization of the COVID-19 Pandemic Impact on a Mobile Network Operator Traffic. In *Proceedings of the 2020 ACM SIGCOMM conference on Internet measurement conference*, 2020.

[29] Athina Markopoulou, Gianluca Iannaccone, Supratik Bhattacharyya, Chen-Nee Chuah, and Christophe Diot. Characterization of failures in an IP Backbone. In *IEEE INFOCOM 2004*, volume 4, pages 2307–2317. IEEE, 2004.

[30] Athina Markopoulou, Gianluca Iannaccone, Supratik Bhattacharyya, Chen-Nee Chuah, Yashar Ganjali, and Christophe Diot. Characterization of failures in an operational ip backbone network. *IEEE/ACM transactions on networking*, 16(4):749–762, 2008.

[31] Debasis Mitra and Qiong Wang. Stochastic Traffic Engineering for Demand Uncertainty and Risk-aware Network Revenue Management. *IEEE/ACM Transactions on networking*, 13(2):221–233, 2005.

[32] NCTA. National upstream peak growth — observed increase in peak consumer usage — observed increase in peak consumer usage, 2020. [Online; accessed 20-May-2020].

[33] Network World. Why didn't COVID-19 break the internet?, 2020. [Online; accessed 20-May-2020].

[34] Bryan Ng, Matthew Hayes, and Winston KG Seah. Developing a traffic classification platform for enterprise networks with SDN: Experiences & lessons learned. In *2015 IFIP Networking Conference (IFIP Networking)*, pages 1–9. IEEE, 2015.

[35] Ramakrishna Padmanabhan, Aaron Schulman, Alberto Dainotti, Dave Levin, and Neil Spring. How to find correlated Internet failures. In *International Conference on Passive and Active Network Measurement*, pages 210–227. Springer, 2019.

[36] Junghun Park, Hsiao-Rong Tyan, and C-C Jay Kuo. Internet traffic classification for scalable QoS provision. In *2006 IEEE International conference on multimedia and expo*, pages 1221–1224. IEEE, 2006.

[37] Paul Poast. Changing the rules of International Relations, April 2020. https://news.uchicago.edu/videos/covid-2025-changing-rules-international-relations-paul-poast.

[38] Rahul Potharaju and Navendu Jain. When the network crumbles: An empirical study of cloud network failures and their impact on services. In *Proceedings of the 4th annual Symposium on Cloud Computing*, pages 1–17, 2013.

[39] Shahbaz Rezaei and Xin Liu. Deep learning for encrypted traffic classification: An overview. *IEEE communications magazine*, 57(5):76–81, 2019.

[40] Matthew Roughan, Subhabrata Sen, Oliver Spatscheck, and Nick Duffield. Class-of-service mapping for QoS: a statistical signature-based approach to IP traffic classification. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pages 135–148, 2004.

[41] SafeGraph. Footprint data. https://www.safegraph.com/covid-19-data-consortium.

[42] Jane M Simmons. Catastrophic failures in a Backbone Network. *IEEE communications letters*, 16(8):1328–1331, 2012.

[43] Rachee Singh, Manya Ghobadi, Klaus-Tycho Foerster, Mark Filer, and Phillipa Gill. RADWAN: Rate Adaptive Wide Area Network. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, SIGCOMM '18.

[44] Lakshminarayanan Subramanian, Ion Stoica, Hari Balakrishnan, and Randy H Katz. OverQoS: offering Internet QoS using Overlays. *ACM SIGCOMM Computer Communication Review*, 33(1):11–16, 2003.

[45] Sean J Taylor and Benjamin Letham. Forecasting at scale. *The American Statistician*, 72(1):37–45, 2018.

[46] Bruno Vidalenc, Laurent Ciavaglia, Ludovic Noirie, and Eric Renault. Dynamic risk-aware routing for OSPF networks. In *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, pages 226–234. IEEE, 2013.

[47] Anh Viet Vu, Jack Hughes, Ildiko Pete, Ben Collier, Yi Ting Chua, Ilia Shumailov, and Alice Hutchings. Turning Up the Dial: the Evolution of a Cybercrime Market Through Set-up, Stable, and Covid-19 Eras. In *Proceedings of the 2020 ACM SIGCOMM conference on Internet measurement conference*, 2020.

[48] Xipeng Xiao and Lionel M Ni. Internet QoS: A big picture. *IEEE network*, 13(2):8–18, 1999.

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