# Uncertainty-Aware Optimization for Network Provisioning and Routing

Yingjie Bi and Ao Tang
School of Electrical and Computer Engineering, Cornell University
yb236@cornell.edu, atang@ece.cornell.edu

***Abstract*—In this paper, we propose a novel approach for the traffic engineering in computer networks with uncertain demand. By utilizing the different flexibility of network reconfiguration and characteristics of traffic demand in different timescales, we solve the network provisioning and routing problems through stochastic optimizations and their second-order cone programming approximations. Numerical illustration shows that our approach can reduce 25% of the network cost compared with the traditional traffic engineering solution.**

## I. Introduction

Traditionally, network operators handle traffic fluctuations in the networks by reserving redundant capacity on each link based on their empirical experience. Naturally, the reservation of capacity has to be quite conservative. As both the mean and variation of the Internet traffic grow, this approach would become less and less cost effective. To lower the operating cost, network operators are looking for better solutions to accommodate the traffic uncertainty and guarantee their service level agreements (SLA) while avoiding over-provisioning.

Most previous researches on traffic engineering with uncertain traffic demands can be mainly categorized into two types: The first is commonly referred to as oblivious routing, a static routing strategy that is robust to traffic change. In the extreme case, [1] showed how to construct an oblivious routing such that the performance of this universal solution is not too far away from optimality under any traffic demand. Obviously, the performance can be improved if we have more information on the traffic demand, and many algorithms have been proposed based on different constraints on the traffic demand, such as the Hose model [2] where the maximum incoming and outgoing traffic of each node is given, or the possible traffic demands are confined by some polyhedron [3], or the demands are divided into common and unexpected scenarios and treated separately during optimization [4]. There are also stochastic approaches such as [5] that maximize the mean of certain traffic engineering objective assuming the traffic demand is subject to some probability distribution. In general, these oblivious routing approaches can guarantee worst-case performance under a large range of traffic conditions, but the incurred cost is relatively high. For example, in the original oblivious routing approach [1], the oblivious ratio is around 2 for typical networks, which means that the oblivious routing can handle only 50% of the traffic demand the best possible routing solution can support.

This work was supported by the Office of Naval Research under N00014-17-1-2419.

The second type of approaches is to develop responsive strategies that update the routing configuration to adapt the realtime traffic demand, such as [6], [7]. Ideally, the timescale of updating should be as fast as possible to make sure that the network configuration is always optimal for the current demand. However, especially in backbone networks, due to factors such as link propagation delay and imperfect demand measurement, routing reconfiguration has to be done at a much slower timescale where all the above factors can be ignored, thus limiting the performance of these dynamic network control algorithms.

In this paper, we propose a novel traffic engineering methodology that combines the ideas from both oblivious routing and dynamic routing[1] by utilizing the different flexibility of network reconfiguration and characteristics of traffic demand in different timescales: (i) At the long-term timescale, typically within the interval of a few months, the network operators can freely reconfigure the network, such as adding or removing links, changing the link capacity or adjusting the routes. On the other hand, the available information on the demand side is very limited in this stage. (ii) At the short-term timescale approximately from an hour to several minutes, since there is no switching between different workdays or between peak and off-peak hours, the seasonal component of traffic variation influenced by human activities does not exist. Thus the traffic demand is much more stable compared with the demand in the long-term timescale. However, at this timescale the reconfiguration of the network is also restricted. For example, it is nearly impossible to change the network topology or link capacities. But with the advance of software-defined network technologies, the routes and split ratio of traffic can still be adjusted to optimize the network performance. (iii) Finally, at the even faster timescale of seconds or below, as we discussed above, the complex transient behaviors forbid any updates to the network configuration. Therefore, even assuming that the traffic demand could be measured instantly (which itself is not an easy job), the most complete information given at this timescale is of little help to the network optimization.

[1]There are other ways to combine the oblivious routing and the dynamic routing, such as [8] which uses oblivious routing to select the available paths and dynamic routing to determine the sending rates on each path.

Roughly speaking, at slower timescales, our approach is closer to the dynamic routing, where we adaptively change the network configuration but do not need to deal with the transient issues. On the contrary, at faster timescales, our approach is closer to the oblivious routing, where we use a single routing solution to handle all the traffic fluctuations. But at this time, the traffic demand is relatively predictable and the needed reserved capacities can be reduced. Since the decision made at slower timescales will affect the available decision region and attainable performance in the faster timescales, we model the traffic engineering problem as a joint optimization across the three timescales.

This paper is organized as follows: We introduce the network and traffic model in Section II and formulate the traffic engineering problem as stochastic optimizations across the above three timescales in Section III. In Section IV, we propose an approximate algorithm to solve the optimization problems by convexifying the original ones. Finally, in Section V, we illustrate the effectiveness of our method using a sample backbone network.

## II. Network and Traffic Model

We consider a network with $L$ uni-directional links in which the capacity of link $l$ is $c_l$. There are $N$ types of traffic demand, and each demand $i$ has $K^i$ available paths from its source to its destination. In this paper, we assume that the set of available paths for the demand is predetermined.[2] The available paths of demand $i$ are represented by an $L \times K^i$ matrix $R^i$, where $R^i_{lk} = 1$ if path $k$ of demand $i$ passes through link $l$ and $R^i_{lk} = 0$ otherwise.

For a given demand $i$, its traffic volume is denoted as $D^i(t)$, modeled as a random process described in the following. The split ratio vector $\alpha^i$ is of size $K^i \times 1$, where the $k$th entry $\alpha^i_k$ is the percentage of traffic that will be routed on its path $k$.

We adopt the temporal traffic model presented in [9]. In this model, the traffic volume $D^i(t)$ is characterized by the following three components:[3]

- The long-term trend $L^i(t)$.
- The seasonal component $S^i(t)$.
- The random fluctuations $W^i(t)$.

The long-term trend $L^i(t)$ captures the overall growth of traffic over a long time period induced by increasing network speed and number of networking devices. The seasonal component $S^i(t)$ is a periodic function influenced by human activities. $W^i(t)$ is a random process of zero mean and unit variance, and the fluctuations of different traffic demands are assumed to be independent from each other. The total traffic volume is given by

$$D^i(t) = L^i(t)S^i(t) + \sqrt{a^i L^i(t) S^i(t)}\, W^i(t), \tag{1}$$

where $a^i$ is a constant called the peakedness of the traffic.

[2] For instance, the SLA may require the network operator to route the demand on paths whose latency is at most 110% of that on the shortest path.

[3] In [9], there is another anomaly component capturing the spikes caused by rare events such as link failures and network attacks, which is outside the scope of this paper.

TABLE I
Characteristics of the Three Timescales in Network Optimization

| Timescale | Typical Granularity | Known Information | Decision Variables |
|---|---|---|---|
| Long-term | A month | $L^i(t)$ | $c_l$, $\alpha^i_k$ |
| Short-term | An hour | $L^i(t)$, $S^i(t)$ | $\alpha^i_k$ |
| Transient | A second | $L^i(t)$, $S^i(t)$, $W^i(t)$ | None |

The three timescales for the network optimization are summarized in Table I. If the value of a random process belongs to the known information of a given timescale, it will be regarded as a constant when we solve the corresponding optimization problem at that timescale. In this paper, $W^i(t)$ is further assumed to be a white Gaussian process. Moreover, due to the slowly varying and periodic behavior of the seasonal component, we can approximate the functions $S^i(t)$ by $Q$ scenarios, i.e.,

$$S^i(t) = S^i_{q(t)}, \quad i = 1, \ldots, N,$$

where $q(t)$ is a periodic and piecewise constant function whose value is an integer between 1 and $Q$. For instance, if the period of $S^i(t)$ is 24 hours, then $Q$ is equal to 24 and $S^i_q$ represents the average traffic volume of demand $i$ on hour $q$.

## III. Stochastic Network Optimization

In this part, we propose the network optimization problem related to each timescale. As assumed above, there is no available control action and thus no optimization problem at the transient timescale. At the short-term timescale, the link capacities $c_l$ are fixed, and the goal of the network operator is to find a routing configuration $\alpha^i_k$ on which the network can run smoothly under traffic fluctuations in the transient timescale. At the long-term timescale, the objective is to design the link capacities $c_l$ minimizing the network cost. On the other hand, it is not necessary to decide the routing configuration $\alpha^i_k$ at this moment since it can be changed at the short-term timescale. Instead, we only need to guarantee the existence of routing configuration $\alpha^i_{k,q}$ for each possible scenario $q$ satisfying the performance requirement.

### A. Short-Term Optimization

We start from the easier problem of optimizing the network at the short-term timescale. Based on our cross-layer optimization methodology, we want to design the routing for the demands such that with a high probability the traffic fluctuations in the transient timescale can be handled using the fixed link capacities that have already been allocated at the long-term timescale. To do this, we first need to model the network behavior at the transient timescale.

For simplicity, we describe the network state at the transient timescale by a quasi-static model that ignores the effect of buffers and propagation delays. At some instant, the traffic volume $D^i$ is a Gaussian random variable whose mean and variance can be obtained from (1) and the given values of the long-term trend $L^i$ and the seasonal component $S^i$. Given the split ratio $\alpha_k^i$, the actual traffic $Y_l$ on link $l$ is

$$Y_l = \sum_{i=1}^{N} \sum_{k=1}^{K^i} R_{lk}^i \alpha_k^i D^i.$$

The short-term network optimization problem is to minimize the probability of SLA violations, which can be defined as the probability of the event that the capacity of some link is exceeded. The complete short-term network optimization problem can be written as

$$\begin{aligned} \min \quad & \Pr\left(\bigcup_{l=1}^{L} \{Y_l > c_l\}\right) \\ \text{s.t.} \quad & \sum_{k=1}^{K^i} \alpha_k^i = 1, \quad \forall i = 1, \dots, N, \\ & \alpha_k^i \geq 0, \quad \forall i = 1, \dots, N,\ k = 1, \dots, K^i. \end{aligned} \tag{2}$$

### *B. Long-Term Optimization*

At the long-term timescale, the only information known to the operator is the long-term traffic trend term $L^i$, and the goal is to choose the link capacities such that the network cost is minimized while the SLA will not be breached in most of the time. However, the split ratios $\alpha_k^i$ do not need to be decided at this moment, because in the later they can be updated to the optimal solution to the short-term optimization problem (2) when the value of the seasonal component $S^i$ becomes known.

Mathematically, let $\epsilon$ be a constant representing the maximum acceptable probability of SLA violation. For each scenario $q$, we have to ensure that there are suitable split ratios $\alpha_{k,q}^i$ such that the corresponding violation probability is bounded by $\epsilon$. Suppose the link capacities have a linear cost function, where $p_l$ is the cost of link $l$ per unit capacity, then the long-term optimization problem can be formulated as

$$\begin{aligned} \min \quad & \sum_{l=1}^{L} p_l c_l \\ \text{s.t.} \quad & \Pr\left(\bigcup_{l=1}^{L} \{Y_{l,q} > c_l\}\right) \leq \epsilon, \quad \forall q = 1, \dots, Q, \\ & \sum_{k=1}^{K^i} \alpha_{k,q}^i = 1, \quad \forall i = 1, \dots, N,\ q = 1, \dots, Q, \\ & \alpha_{k,q}^i \geq 0, \quad \forall i = 1, \dots, N,\ k = 1, \dots, K^i, \\ & \qquad q = 1, \dots, Q, \end{aligned} \tag{3}$$

where

$$Y_{l,q} = \sum_{i=1}^{N} \sum_{k=1}^{K^i} R_{lk}^i \alpha_{k,q}^i D_q^i$$

and $D_q^i$, the traffic volume in scenario $q$, is a Gaussian random variable whose mean $d_{i,q}$ and variance $\sigma_{i,q}^2$ are determined by (1) and $S_q^i$. Hence $Y_{l,q}$ is also Gaussian and its mean $y_{l,q}$ and variance $b_{l,q}$ are given by

$$\begin{aligned} y_{l,q} &= \mathrm{E}[Y_{l,q}] = \sum_{i=1}^{N} \sum_{k=1}^{K^i} R_{lk}^i \alpha_{k,q}^i d_{i,q}, \\ b_{l,q} &= \mathrm{Var}[Y_{l,q}] = \sum_{i=1}^{N} \left(\sum_{k=1}^{K^i} R_{lk}^i \alpha_{k,q}^i\right)^2 \sigma_{i,q}^2. \end{aligned} \tag{4}$$

## IV. Approximate Solutions to Stochastic Network Optimization

The probability term poses significant difficulty in solving the short-term (2) and long-term (3) problem, and we have to resort to approximate solutions.

Let us first consider the long-term problem (3). Note that in practice the upper bound $\epsilon$ for the violation probability is small, so the probability of the event that the capacity of two or more links is exceeded can be safely ignored. Therefore, we can use

$$\sum_{l=1}^{L} \Pr(Y_{l,q} > c_l) \leq \epsilon$$

to replace the original probability constraints in (3) without increasing much cost. To further simplify the problem, we constraint the violation probability of each link separately by setting

$$\Pr(Y_{l,q} > c_l) \leq \epsilon / L, \quad l = 1, \dots, L. \tag{5}$$

Then the probability constraints in (3) will be automatically satisfied as long as all the above constraints (5) hold.

Let $f(x)$ be the survival function of standard normal distribution, i.e.,

$$f(x) = \Pr(X > x),$$

where $X$ is subject to the standard normal distribution. Using (4) to normalize the random variable $Y_{l,q}$, we get

$$\begin{aligned} \epsilon / L &\geq \Pr(Y_{l,q} > c_l) \\ &= \Pr\left(\frac{Y_{l,q} - y_{l,q}}{\sqrt{b_{l,q}}} > \frac{c_l - y_{l,q}}{\sqrt{b_{l,q}}}\right) \\ &= f\left(\frac{c_l - y_{l,q}}{\sqrt{b_{l,q}}}\right). \end{aligned}$$

Define

$$A = \frac{1}{f^{-1}(\epsilon / L)}.$$

As $f(x)$ is monotonically decreasing, (5) is equivalent to

$$\frac{c_l - y_{l,q}}{\sqrt{b_{l,q}}} \geq \frac{1}{A}.$$

Combining the above inequality with (4), we approximate the original long-term problem (3) as follows:

$$\begin{aligned}
\min \quad & \sum_{l=1}^{L} p_l c_l \\
\text{s.t.} \quad & \sum_{i=1}^{N} \left( \sum_{k=1}^{K^i} R_{lk}^i \alpha_{k,q}^i \right)^2 \sigma_{i,q}^2 \leq A^2 (c_l - y_{l,q})^2, \\
& \quad \forall l = 1, \ldots, L,\ q = 1, \ldots, Q, \\
& y_{l,q} = \sum_{i=1}^{N} \sum_{k=1}^{K^i} R_{lk}^i \alpha_{k,q}^i d_{i,q} \leq c_l, \quad \forall l = 1, \ldots, L, \\
& \quad q = 1, \ldots, Q, \\
& \sum_{k=1}^{K^i} \alpha_{k,q}^i = 1, \quad \forall i = 1, \ldots, N,\ q = 1, \ldots, Q, \\
& \alpha_{k,q}^i \geq 0, \quad \forall i = 1, \ldots, N,\ k = 1, \ldots, K^i, \\
& \quad q = 1, \ldots, Q.
\end{aligned} \tag{6}$$

The above problem (6) is in the form of second-order cone programming, which is a well-studied convex optimization problem that can be efficiently solved. Next, we will analyze the performance loss of the approximate solution obtained from (6) from the actual optimal solution for (3).

*Theorem 1:* Let opt be the optimal value of the original problem (3) and sol be the optimal value of the approximation (6), then

$$\text{opt} \leq \text{sol} \leq \frac{A'}{A} \text{opt},$$

where

$$A' = \frac{1}{f^{-1}(\epsilon)}.$$

*Proof:* Since the optimal solution to the approximation (6) is feasible for the original problem (3), we have opt $\leq$ sol. For the other direction, pick up an optimal solution to (3), and let $\bar{c}_l$ be the chosen link capacities in this solution. Then for each link $l$,

$$\Pr(Y_{l,q} > \bar{c}_l) \leq \Pr\left( \bigcup_{l'=1}^{L} \{Y_{l',q} > \bar{c}_{l'}\} \right) \leq \epsilon.$$

After normalizing the random variable $Y_{l,q}$, we get

$$\frac{\bar{c}_l - y_{l,q}}{\sqrt{b_{l,q}}} \geq \frac{1}{A'}.$$

Define $\hat{c}_l = \bar{c}_l A'/A$. Since $A \leq A'$ and $y_{l,q} \geq 0$,

$$\frac{\hat{c}_l - y_{l,q}}{\sqrt{b_{l,q}}} \geq \frac{A'}{A} \frac{\bar{c}_l - y_{l,q}}{\sqrt{b_{l,q}}} \geq \frac{1}{A}.$$

Using the capacities $\hat{c}_l$ and the same split ratios from the considered optimal solution, by the above inequality we get a feasible solution to the approximate problem (6) whose objective value

$$\frac{A'}{A} \text{opt} \geq \text{sol},$$

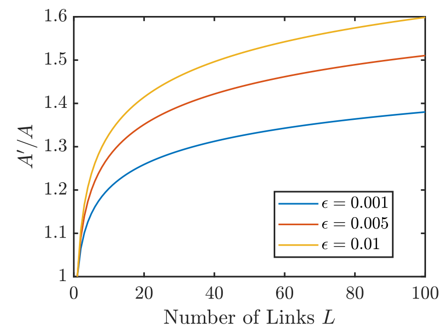



Fig. 1. The relationship between the approximation ratio $A'/A$ of proposed approximation algorithm and the number of links $L$ in the network for different violation probability $\epsilon$.

which completes the proof. ∎

To study the approximation ratio of the approximation algorithm, we need to understand how $A'/A$ depends on $\epsilon$ and the number of links $L$. The following property about the survival function $f(x)$ will be helpful:

*Lemma 2:* If $x > 0$,

$$\left( \frac{1}{x} - \frac{1}{x^3} \right) \frac{1}{\sqrt{2\pi}} e^{-x^2/2} < f(x) < \frac{1}{x} \frac{1}{\sqrt{2\pi}} e^{-x^2/2}.$$

For its proof, see [10, Section 7.1]. If $x > 2$, the above inequality implies

$$\frac{1}{x+1} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} < f(x) < \frac{1}{\sqrt{2\pi}} e^{-x^2/2}. \tag{7}$$

Suppose $\epsilon$ is sufficiently small such that $A \leq A' < 1/2$. Applying the inequality (7), we obtain

$$\frac{\epsilon}{L} = f\left( \frac{1}{A} \right) < \frac{1}{\sqrt{2\pi}} e^{-1/2A^2}.$$

Thus

$$\frac{1}{2A^2} < \log L - \log(\sqrt{2\pi}\epsilon). \tag{8}$$

On the other hand,

$$\epsilon = f\left( \frac{1}{A'} \right) > \frac{1}{1/A' + 1} \frac{1}{\sqrt{2\pi}} e^{-1/2A'^2}.$$

By the inequality (7) again,

$$\begin{aligned}
\frac{1}{2A'^2} &> -\log(1/A' + 1) - \log(\sqrt{2\pi}\epsilon) \\
&\geq -1/A' - \log(\sqrt{2\pi}\epsilon) \\
&\geq -1/A'^2 - \log(\sqrt{2\pi}\epsilon),
\end{aligned}$$

so

$$\frac{3}{2A'^2} > -\log(\sqrt{2\pi}\epsilon). \tag{9}$$

Combining the both inequalities (8) and (9) above, we get

$$A'/A \leq O(\sqrt{\log L / \log(1/\epsilon)}).$$

The above analysis provides the order of approximation ratio as the number of links $L$ increasing. In fact, for fixed $L$ and $\epsilon$, we can directly compute $A'/A$, which gives the upper bound for the approximation ratio in all networks containing at most $L$ links. Fig. 1 shows the relationship between $A'/A$ and $L$ for different violation probability $\epsilon$. In general, the performance of the approximation algorithm decreases slowly when the number of links $L$ increases, and the performance would be better if $\epsilon$ is smaller. For typical backbone networks in which $L$ is around 20, the approximation ratio is in the range of 1.2–1.4, which means that our algorithm is guaranteed to find a feasible solution whose cost is at most 40% higher than the actual optimal cost.

Similar idea can also be applied to the short-term optimization problem (2). First, we introduce a new decision variable $\epsilon$ for problem (2) and rewrite this problem equivalently by changing the objective function to be $\epsilon$ and adding a new constraint

$$\Pr\left(\bigcup_{l=1}^{L}\{Y_l > c_l\}\right) \le \epsilon.$$

By the same steps for the long-term problem (3), the problem (2) can also be approximated as

$$\begin{aligned} \min \quad & A \\ \text{s.t.} \quad & \sum_{i=1}^{N}\left(\sum_{k=1}^{K^i} R_{lk}^i \alpha_k^i\right)^2 \sigma_i^2 \le A^2(c_l - y_l)^2, \\ & \qquad \forall l = 1,\dots,L, \\ & y_l = \sum_{i=1}^{N}\sum_{k=1}^{K^i} R_{lk}^i \alpha_k^i d_i \le c_l, \quad \forall l = 1,\dots,L, \\ & \sum_{k=1}^{K^i} \alpha_k^i = 1, \quad \forall i = 1,\dots,N, \\ & \alpha_k^i \ge 0, \quad \forall i = 1,\dots,N,\ k = 1,\dots,K^i, \end{aligned} \tag{10}$$

where $d_i$ and $\sigma_i^2$ are the mean and variance of the traffic volume $D^i$, respectively. For fixed $A$, the constraints in the above problem (10) are all linear or second-order cone constraints, so we can find the optimal solution to (10) by bisection searching over $A$, which provides an approximation algorithm to solve the original short-term optimization problem (2) with approximation ratio $L$.

## V. Numerical Example

To illustrate the effectiveness of our new approach, in this part we will apply our method to the Abilene network as an example, which has 11 nodes and 15 undirected links (see [9, Fig. 2(a)] for its topology). To be consistent with the model in this paper, we assume that all links can be used to transmit in either direction and the capacities for both directions can be chosen independently. We also assume that there is a traffic demand for every pair of nodes in the network. Two available paths are picked for each demand, which are the shortest and the second shortest path between the source and destination in terms of the number of hops. The parameters in the traffic model (1) are determined by the following way: The long-term trend $L^i$ is randomly chosen subject to uniform distribution in $[1.5, 10]$. There will be $Q$ scenarios for the short-term timescale, and the seasonal component $S_q^i$ in each scenario $q$ is also uniformly randomly chosen in $[1, 1.5]$. The random fluctuation $W^i$ in the transient timescale is standard normal, and the peakedness $a^i$ in (1) all equals to a constant $a$. Furthermore, for simplicity the link cost $p_l$ per unit capacity for each link $l$ is assumed to be 1.

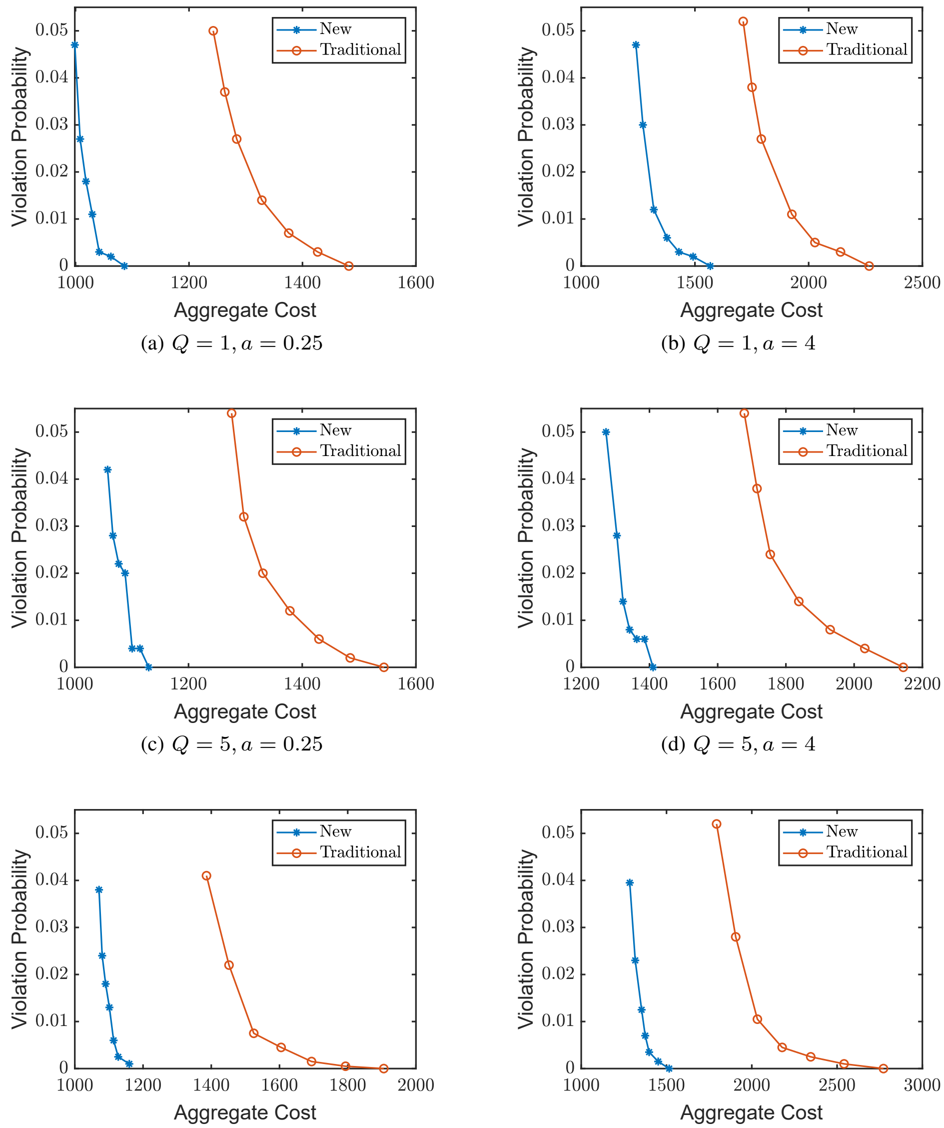
We compare our result with the traditional approach [11] in which network operators handle the traffic fluctuations by directly limiting the link utilization. One possible formulation is as follows:

$$\begin{aligned} \min \quad & \sum_{l=1}^{L} p_l c_l \\ \text{s.t.} \quad & \sum_{i=1}^{N}\sum_{k=1}^{K^i} R_{lk}^i \alpha_k^i d_i \le \rho c_l, \quad \forall l = 1,\dots,L, \\ & \sum_{k=1}^{K^i} \alpha_k^i = 1, \quad \forall i = 1,\dots,N, \\ & \alpha_k^i \ge 0, \quad \forall i = 1,\dots,N,\ k = 1,\dots,K^i. \end{aligned} \tag{11}$$

Here $d_i$ is the estimated average traffic volume for demand $i$, while $0 < \rho \le 1$ is a constant representing the maximum acceptable link utilization specified based on the experience of the operator. In the optimal solution to (11), all the link capacity constraints must be tight and thus the utilization of each link is exactly $\rho$. However, the actual fluctuations of flow rate on links can be quite different, and links with smaller fluctuations have reserved unnecessary extra capacity. In contrast, our method is able to figure out the precise capacity needed to guarantee the SLA requirements, saving huge cost compared with the traditional approach.

Now we evaluate the performance for both approaches. For the traditional approach, we solve the problem (11) with $d_i$ being the mean value of traffic $D^i$. For our new approach, we solve the long-term problem (6) first to determine the link capacities $c_l$ and then solve the short-term problem (10) for each scenario $q$ to determine the split ratios $\alpha_{k,q}^i$. Next, a large number of cases of traffic demand are generated for all the scenarios according to the assumed probability distribution, and we count the total number of cases in which the capacity of some link is exceeded, giving the empirical link capacity violation probability.

By varying $\rho$ in (11) and $A$ in (6), we obtain a series of solutions with different cost and link capacity violation probability, which are shown in Fig. 2. To minimize the cost, the network operator would choose the maximum $\rho$ or $A$ as long as the corresponding violation probability is acceptable. To achieve the same violation probability (for example, 0.005), the cost of the solution from our new approach is at least 25% lower than the cost from the traditional approach, and the improvement is more significant when the number of scenarios $Q$ or the peakedness $a$ of the traffic in the transient timescale increases.



(a) $Q = 1, a = 0.25$ (b) $Q = 1, a = 4$

(c) $Q = 5, a = 0.25$ (d) $Q = 5, a = 4$

(e) $Q = 20, a = 0.25$ (f) $Q = 20, a = 4$

Fig. 2. The trade-off between the network cost and the link capacity violation probability when the traditional approach or our proposed approach is applied for different number of scenarios $Q$ and different peakedness $a$ of the traffic.

This paper opens up new directions for the traffic engineering problem with uncertain demands, although the quantitative gain depends on the Gaussian assumption of the traffic fluctuations. Future investigations are necessary to pin down the exact statistical distribution using empirical data and to see how our proposed algorithms would behave in real networks.

## References

[1] D. Applegate and E. Cohen, "Making intra-domain routing robust to changing and uncertain traffic demands: Understanding fundamental tradeoffs," in *Proc. ACM SIGCOMM*, Aug. 2003, pp. 313–324.

[2] M. Kodialam, T. V. Lakshman, J. B. Orlin, and S. Sengupta, "Oblivious routing of highly variable traffic in service overlays and IP backbones," *IEEE/ACM Trans. Netw.*, vol. 17, no. 2, pp. 459–472, Apr. 2009.

[3] W. Ben-Ameur and H. Kerivin, "Routing of uncertain traffic demands," *Optim. Eng.*, vol. 6, no. 3, pp. 283–313, Sep. 2005.

[4] H. Wang, H. Xie, L. Qiu, Y. R. Yang, Y. Zhang, and A. Greenberg, "COPE: Traffic engineering in dynamic networks," in *Proc. ACM SIGCOMM*, Sep. 2006, pp. 99–110.

[5] D. Mitra and Q. Wang, "Stochastic traffic engineering for demand uncertainty and risk-aware network revenue management," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 221–233, Apr. 2005.

[6] M. Alizadeh *et al.*, "CONGA: Distributed congestion-aware load balancing for datacenters," in *Proc. ACM SIGCOMM*, Aug. 2014, pp. 503–514.

[7] J. Perry, A. Ousterhout, H. Balakrishnan, D. Shah, and H. Fugal, "Fastpass: A centralized "zero-queue" datacenter network," in *Proc. ACM SIGCOMM*, Aug. 2014, pp. 307–318.

[8] P. Kumar *et al.*, "Semi-oblivious traffic engineering: The road not taken," in *Proc. USENIX NSDI*, Apr. 2018, pp. 157–170.

[9] P. Tune and M. Roughan, "Internet traffic matrices: A primer," in *Recent Advances in Networking*, H. Haddadi and O. Bonaventure, Eds. ACM SIGCOMM eBook, 2013, vol. 1, pp. 108–163.

[10] W. Feller, *An Introduction to Probability Theory and Its Applications*, 3rd ed. New York, NY, USA: Wiley, 1968, vol. 1.

[11] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 1992.