2011 31st International Conference on Distributed Computing Systems

One More Weight is Enough: Toward the Optimal Traffic Engineering with OSPF

Ke Xu†, Hongying Liu‡, Jiangchuan Liu§, Meng Shen†
Tsinghua University† Beihang University‡ Simon Fraser University§
{xuke,shenmeng}@csnet1.cs.tsinghua.edu.cn liuhongying@buaa.edu.cn jcliu@cs.sfu.ca

***Abstract*—Traffic Engineering (TE) leverages information of network traffic to generate a routing scheme optimizing the traffic distribution so as to advance network performance. However, optimizing the link weights for OSPF to the offered traffic is an known NP-hard problem. In this paper, we model the optimal TE as the utility maximization of multi-commodity flows and theoretically prove that any given set of optimal routes corresponding to a particular objective function can be converted to shortest paths with respect to a set of positive link weights, which can be *explicitly* formulated using the optimal distribution of traffic and objective function. This can be directly configured on OSPF-based protocols. On these bases, we employ the Network Entropy Maximization (NEM) framework and develop a new OSPF-based routing protocol, SPEF, to realize a flexible way to split traffic over shortest paths in a distributed fashion. Actually, comparing to OSPF, SPEF only needs one more weight for each link and provably achieves optimal TE. Numerical experiments have been done to compare SPEF with the current version of OSPF, showing the effectiveness of SPEF in terms of link utilization and network load distribution.**

***Index Terms*—Traffic engineering, OSPF, Utility, Routing**

I. INTRODUCTION

The primary role of Internet Service Providers (ISPs) is to guarantee service via deploying infrastructures, managing network connectivity and balancing traffic load inside their networks [9]. The goal of Traffic Engineering (TE) is to ensure efficient routing to minimize network congestion, so that users can experience low packet loss, high throughput, and low latency. Traffic Engineering leverages information from traffic entering and leaving the network to generate a routing scheme that optimizes network performance. In particular, an ISP solves the TE problem by adjusting the routing configuration to the prevailing traffic.

In this paper, we focus on traffic engineering within a single Autonomous Systems (AS), in which we assume that the egress point of each external destination is known and fixed. Traffic engineering thus depends on a set of performance objectives that guide path selection, as well as effective mechanisms for routers to select paths that satisfy these objectives [15].

Open Shortest Path First (OSPF) is a commonly used intra-domain routing protocol [23], which provides the network operators a way to control network routing by configuring OSPF link weights. The quality of OSPF-based traffic engineering depends largely on the choice of weights. Link weights can have a reasonable default configuration based on link capacity, *e.g.*, Cisco's InvCap [10] sets the weight of a link inversely proportional to its capacity, which can be explained by the $M/M/1$ queuing model. Although fairly intuitive and convenient, these setting approaches might lead to undesirable network load distribution, since they do not take the expected traffic demand into consideration. In practice, given network link capacities and expected traffic demands, the link weights can be optimized by ISPs according to a certain object function. However, computing the optimal link weights under the evenly traffic splitting scheme has been proven to be NP-complete [12].

Challenges. In this paper, we take an important step towards building an OSPF-based routing protocol that can achieve the optimal traffic engineering. Although this optimization problem has attracted a great research interests and been extensively studied (e.g., [14], [15], [19]), there are still several challenges to be further studied, including the following:

1. **Can we guarantee the universal existence of optimal link weights?** Network providers are usually interested in various indicators to improve the network performance in different ways, e.g.,some of them might prefer to lower the maximum link utilization, while others might try to minimize path lengths. Accordingly, various objective functions have been proposed to capture these demands. Based on the results derived from linear programming, Wang et al. [15] proved that any arbitrary set of routes can be converted to shortest-paths with respect to some set of positive link weights. Although this outcome is encouraging to some extent, we still prefer to ensure the universal existence of optimal link weights which are *explicitly* determined by the objective function and the optimal distribution of traffic.

2. **Can we achieve the optimal TE for intra-domain IP networks based on OSPF?** As a distributed link-state routing protocol, OSPF uses the shortest path routing with destinations based hop-by-hop forwarding and Equal-Cost Multi-Path (ECMP) mechanism to evenly split the corresponding traffic over all available equal-cost paths. Many approaches are proposed attempting to achieve the "optimal" routing based on OSPF. Wang et al. [15] and Srivastava et al. [14] proposed flexible solutions to efficiently split traffic over shortest paths,

This work has been supported in part by NSFC Project(60970104), 973 Project of China (2009CB320501), 863 Project of China (2008AA01A323), 863 Project of China (2008AA01A326) and Program for New Century Excellent Talents in University.

1063-6927/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDCS.2011.49

but these centralized solutions went against the distributed feature of OSPF. A new link-state protocol named PEFT, recently proposed by Xu et al. [16], successfully realized a flexible traffic splitting scheme in a distributed manner, whereas failed to maintain the shortest paths in packet forwarding thus sacrificing a key benefit of OSPF. Guaranteeing the crucial features of OSPF in terms of scalability and efficiency are thus a great challenge in achieving the optimal traffic engineering goals based on OSPF.

Our Approach and Contributions. In this paper, we model the optimal TE as the utility maximization of multi-commodity flows and propose a distributed dual decomposition method to compute the optimal link weights. Based on these, we develop a new OSPF-based protocol, Shortest paths Penalizing Exponential Flow-splitting (SPEF). It has been proved able to achieve the optimal TE.

Toward the optimal TE, in SPEF, we only need one more weight for each link. For simplicity, we hereafter refer to the optimal and the additional link weights as the first and second link weights, respectively. In SPEF, packets forwarding is the same as OSPF: hop-by-hop along the shortest paths constructed based on destination according to the first link weights. When there are multiple shortest paths for the same source-destination pairs from the perspective of the first link weights, the flow split ratio over these multiple shortest paths can be independently computed by routers using the second link weights. In particular, we address the above challenges as follows:

1. To ensure the existence of optimal link weights, we model the optimal TE as the utility maximization of multi-commodity flows and theoretically show that any given set of optimal routes corresponding to a particular objective function can be converted to shortest paths with respect to a set of positive link weights, which can be *explicitly* formulated using the optimal distribution of traffic and objective function. Inspired by the rate control scheme in congestion control [7], we investigate a class of (q, β) utility function as the objective function to extract various demands in TE.

2. To achieve the optimal TE based on OSPF, we develop a new routing protocol, SPEF, proving that it can achieve the optimal TE for intra-domain IP networks. Although we leverage the NEM framework proposed in PEFT, the key difference between them is in that: SPEF realizes a flexible flow splitting over shortest paths in a distributed fashion, guaranteeing the crucial features of OSPF in terms of scalability and efficiency.

Paper Organization. The rest of the paper is organized as follows. We present the utility maximization of multi-commodity flows and theoretically prove the universal existence of the optimal link weights in Section II. In Section III, we propose the (q, β) utility function to extract various needs in TE and then explore characteristics of the utility maximization problem with the logarithmic utility objective function. The new OSPF-based protocol is developed in Section IV, following which is the performance evaluation in Section V. Related work is summarized in Section VI, before we conclude with the achievements and extensions in Section VII.

II. Optimal Weights Existence

In this section, we first present the network model and resort to the utility maximization of multi-commodity flows to model the optimal TE. Then we theoretically prove that the optimal link weights always exist under the generic objective functions with different parameters.

A. Network Model

We consider a directed network $\mathcal{G} = (\mathcal{N}, \mathcal{J})$ with vertex set $\mathcal{N}$, edge set $\mathcal{J}$, and R source-destination vertex pairs $\{s_1, t_1\}, \cdots, \{s_R, t_R\}$. Each edge (i, j) has a capacity c_{ij}, which is a measure for the amount of traffic flow it can take. A demand (traffic) for (s_r, t_r) is d_r, which denotes the average intensity of traffic entering the network at vertex s_r and exiting at vertex t_r. In the following, we use notations N, J to denote the cardinalities of sets $\mathcal{N}$ and $\mathcal{J}$ respectively and $\mathcal{R} = \{1, \cdots, R\}$ to denote the source-destination vertex pairs index set.

The multi-commodity flow problem is a network flow problem with multiple commodities (or goods) flowing through the network, with different source and sink nodes. The more customary way to treat routing in a network is to consider it as a multi-commodity flow problem. Denote the destination node set with $\mathcal{D} = \{t \in \mathcal{N} : \exists\ r \in \mathcal{R} \text{ s.t. } t_r = t\}$. The traffic flow to each destination $t \in \mathcal{D}$ can be regarded as a commodity. The flow of commodity t along edge (i, j) is f_{ij}^t. Find an assignment of flow satisfying the constraints:

$$f_{ij} = \sum_{t \in \mathcal{D}} f_{ij}^t \leq c_{ij}, \quad \forall (i, j) \in \mathcal{J} \tag{1a}$$

$$\sum_{j:(s,j)\in\mathcal{J}} f_{sj}^t - \sum_{i:(i,s)\in\mathcal{J}} f_{is}^t = d_s^t, \forall t \in \mathcal{D}, \forall s \in \mathcal{N} \backslash \{t\}, \tag{1b}$$

$$f_{ij}^t \geq 0,\ \forall t \in \mathcal{D}, (i, j) \in \mathcal{J}, \tag{1c}$$

where (1a) and (1b) are the capacity constraints and flow conservation constraints, respectively, and $d_s^t \geq 0$ is the expected traffic entering the network at node s and destined to node t. Set $d_s^t = d_r$ if there exists $r \in \mathcal{R}$ such that $s_r = s$ and $t_r = t$, or set $d_s^t = 0$ otherwise.

We say a traffic distribution $\mathbf{f} = (f_{ij}, (i, j) \in \mathcal{J})$ is feasible if there exists $(\mathbf{f}^t, t \in \mathcal{D})$ such that $(\mathbf{f}, \mathbf{f}^t, t \in \mathcal{D})$ satisfies the multi-commodity flow constraints (1). If $\mathbf{f}$ is feasible, the total load on and the utilization of the link $(i, j) \in \mathcal{J}$ are f_{ij} and $\frac{f_{ij}}{c_{ij}}$ respectively, which depend on how the network decides to route the traffic. Now, one main task is to find an *appropriate* and *feasible* traffic distribution $\mathbf{f}$.

An objective function enables quantitative comparisons between different routing solutions in terms of load f_{ij} on the links. Traffic engineering usually considers a link-cost function $\Phi(\mathbf{f}, \mathbf{c})$ that is an increasing function of $\mathbf{f}$. *Optimal traffic engineering* [11] means that the TE cost function is minimized over multi-commodity flow constraints (1).

B. Utility Model of Traffic Engineering

For the offered traffic, TE changes routing to minimize network congestion. Here we use the utility maximization solution to route traffic, which is equal to the multi-commodity flow solution. The reason is two-fold: (a) it is optimal, *i.e.*, it guarantees the routing with maximum spare capacity utility; (b) it can be realized by routing protocols that use MPLS tunneling, or in a distributed fashion by OSPF routing.

We associate link (i,j) with an operator, and assume that if a spare capacity s_{ij} is held by operator (i,j), which has utility $V_{ij}(s_{ij})$ to the operator. We assume that the utility $V_{ij}(s_{ij})$ is an increasing, concave and continuously differentiable function of s_{ij} over the range $s_{ij} \geq 0$, and $V'_{ij}(s_{ij}) > 0$ over the range $s_{ij} \geq 0$. Assume further that utilities are additive, so that the aggregate utility of spare capacity $\mathbf{s} = (s_{ij}, (i,j) \in \mathcal{J})$ is $\sum_{(i,j)\in\mathcal{J}} V_{ij}(s_{ij})$.

It is the concavity of the function V_{ij} that balances traffic distribution among links. If V_{ij} is a convex increasing function instead of a concave, then maximize the aggregate utility. Larger spare capacity s_{ij} should be increased, since the rate of increase of V_{ij} is increasing in s_{ij}. When V_{ij} is linear, the rate of increase of V_{ij} is the same for all s_{ij}. When V_{ij} is concave, a smaller spare capacity s_{ij} is preferred, since $V'_{ij}(x) > V'_{ij}(y)$ holds if $x < y$.

Now the optimal traffic engineering can be formulated as maximizing the aggregated utility under the multi-commodity flow constraints (1).

TE$(V, \mathcal{G}, \mathbf{c}, \mathbf{D})$

$$\text{maximize}_{\mathbf{f}^t \geq \mathbf{0}} \sum_{(i,j)\in\mathcal{J}} V_{ij}(s_{ij}) \tag{2a}$$

$$\text{subject to} \quad \mathbf{c} - \sum_{t\in\mathcal{D}} \mathbf{f}^t = \mathbf{s} \geq \mathbf{0} \tag{2b}$$

$$\mathbf{B}\mathbf{f}^t = \mathbf{d}^t, \ \forall t \in \mathcal{D}, \tag{2c}$$

where $\mathbf{B}$, an $N \times J$ node-arc incidence matrix for network $\mathcal{G}$, is introduced to represent the multi-commodity flow constraints (1). The j-th column of $\mathbf{B}$ corresponding to link $(u,v) \in \mathcal{J}$ is defined as

$$B_{ij} = \begin{cases} 1, & i = u \\ -1, & i = v \\ 0, & \text{otherwise,} \end{cases}$$

There is a unique optimum for the spare capacity vector $\mathbf{s}$, since the objective function (2a) is a strictly concave function of $\mathbf{s}$. But there may be many values of the flow vector $(\mathbf{f}^t, t \in \mathcal{D})$ satisfying relations (2b) and (2c). Say that $\mathbf{s}$ solves TE$(V, \mathcal{G}, \mathbf{c}, \mathbf{D})$ if there exists $(\mathbf{f}^t, t \in \mathcal{D})$ such that $(\mathbf{s}, \mathbf{f}^t, t \in \mathcal{D})$ solves the optimization problem (2).

C. Optimal Weights Existence

From the general theory of constrained convex optimization [2], it follows that $(\mathbf{s}, \mathbf{f}^t, t \in \mathcal{D})$ solves problem (2) if and only if there exists Lagrangian multiplier vectors $\mathbf{w}$ and $\boldsymbol{\nu}^t, t \in \mathcal{D}$ that satisfy

$$\mathbf{c} - \sum_{t\in\mathcal{D}} \mathbf{f}^t = \mathbf{s}, \quad \mathbf{B}\mathbf{f}^t = \mathbf{d}^t, \ \forall t \in \mathcal{D} \tag{3a}$$

$$V'_{ij}(s_{ij}) - w_{ij} = 0, \quad \text{if} \quad s_{ij} > 0 \tag{3b}$$

$$\leq 0, \quad \text{if} \quad s_{ij} = 0 \tag{3c}$$

$$\nu^t_j - \nu^t_i - w_{ij} = 0, \quad \text{if} \quad f^t_{ij} > 0 \tag{3d}$$

$$\leq 0, \quad \text{if} \quad f^t_{ij} = 0. \tag{3e}$$

We define $c'_{ij} = c_{ij} - s_{ij}$ as the *target capacity* for each link, which is no greater than the actual capacity c_{ij} (a "virtual" capacity). This is also desirable since it leads to an empty equilibrium.

From $V'_{ij}(s_{ij}) > 0$ as well as Eq. (3b) and (3c), we have $w_{ij} > 0$. The Lagrangian multiplier vectors $\mathbf{w}$ and $\boldsymbol{\nu}^t$ have several simple interpretations. Let $p : i_0 i_1 i_2 \cdots i_m$ be a possible path of source-destination pair (s,t), where $i_0 = s$ and $i_m = t$. For example, if $y_p = \min_{k=1,2,\cdots,m} f^t_{i_{k-1}i_k} > 0$, we have $\sum_{(i,j)\in p} w_{ij} = \nu^t_t - \nu^t_s \leq \sum_{(i,j)\in\bar{p}} w_{ij}$ for any other path $\bar{p}$ that connects the same source-destination pair (s,t) under the conditions (3d) and (3e). We may view w_{ij} as the implied cost of traffic through link (i,j). Alternatively, w_{ij} is the *shadow price* of additional capacity at link (i,j). We can also regard $\mathbf{w}$ as the weight set by the operator.

Let $\mathbf{w} = (w_{ij} : (i,j) \in \mathcal{J})$ and $(\mathbf{s}, \mathbf{f}^t, t \in \mathcal{D})$ be a solution of (2). We have shown that $\mathbf{f}^t$ determines the shortest path for each source-destination pair (s,t) under the link weights $\mathbf{w}$, which is determined explicitly by the utility function V_{ij} and the spare capacity s_{ij} through Eq. (3b) and (3c).

If link (i,j) is charged price of per unit spare capacity, and is capable to freely vary the spare capacity s_{ij}, then the utility maximization problem for link (i,j) becomes

Link$_{ij}(V_{ij}; w_{ij})$

$$\begin{array}{ll} \text{maximize} & V_{ij}(s_{ij}) - w_{ij}s_{ij} \\ \text{subject to} & s_{ij} \geq 0. \end{array} \tag{4}$$

If the network receives a revenue w_{ij} per unit spare capacity from link (i,j), and is allowed to freely vary the spare capacity s_{ij}, then the revenue optimization problem for the network is as follows.

Network$(\mathcal{G}, \mathbf{c}, \mathbf{D}; \mathbf{w})$

$$\begin{array}{ll} \text{maximize}_{\mathbf{f}^t \geq \mathbf{0}} & \sum_{(i,j)\in\mathcal{J}} w_{ij}s_{ij} \\ \text{subject to} & \mathbf{c} - \sum_{t\in\mathcal{D}} \mathbf{f}^t = \mathbf{s} \geq \mathbf{0} \\ & \mathbf{B}\mathbf{f}^t = \mathbf{d}^t, \ \forall t \in \mathcal{D}. \end{array} \tag{5}$$

$\mathbf{s}$ solves Network$(\mathcal{G}, \mathbf{c}, \mathbf{D}; \mathbf{w})$ if there exists $(\mathbf{f}^t, t \in \mathcal{D})$ such that $(\mathbf{s}, \mathbf{f}^t, t \in \mathcal{D})$ solves the problem (5). Reducing the spare capacity $\mathbf{s}$ from (5), we have that Network$(\mathcal{G}, \mathbf{c}, \mathbf{D}; \mathbf{w})$ is a minimum cost multi-commodity flow problem [3], *i.e.*

$$\begin{array}{ll} \text{minimize}_{\mathbf{f}^t \geq \mathbf{0}} & \sum_{(i,j)\in\mathcal{J}} w_{ij} \sum_{t\in\mathcal{D}} f^t_{ij} \\ \text{subject to} & \sum_{t\in\mathcal{D}} f^t_{ij} \leq c_{ij}, \forall (i,j) \in \mathcal{J} \\ & \mathbf{B}\mathbf{f}^t = \mathbf{d}^t, \ \forall t \in \mathcal{D}. \end{array} \tag{6}$$

Theorem 1 (weight-setting): There exists a weight vector $\mathbf{w} = (w_{ij}, (i,j) \in \mathcal{J})$ such that the vector $\mathbf{s} =$

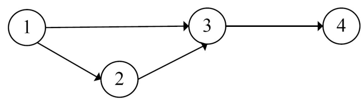


Fig. 1. An simple network topology

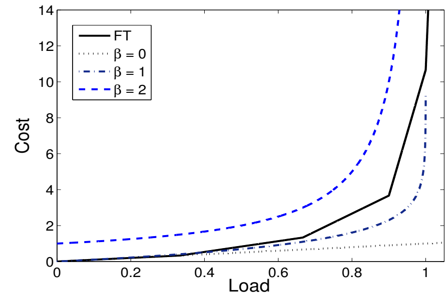


Fig. 2. Different link cost as a function of the load for a link capacity 1, where FT denotes the one proposed by Fortz and Thorp [11] and $q_{ij} = 1$ in (7)

$(s_{ij}, (i,j) \in \mathcal{J})$, formed from the unique solution s_{ij} to $\text{Link}_{ij}(V_{ij}; w_{ij})$, solves $\text{Network}(\mathcal{G}, \mathbf{c}, \mathbf{D}; \mathbf{w})$. The vector $\mathbf{s}$ also solves $\text{TE}(V, \mathcal{G}, \mathbf{c}, \mathbf{D})$.

As the space is limited, please see our extended version [22] for the proofs of Theorem 1 and other theorems hereafter.

We now examine the engineering implications of Theorem 1. It is true that, the Lagrangian multiplier vector $(w_{ij}, (i,j) \in \mathcal{J})$ gives link weights such that all the traffic flow will be forwarded along the minimum cost multi-commodity problem solution. Meanwhile the link (i,j) maximizes its utility through retaining a proper spare capacity. Inversely, if there exists link weights $(w_{ij}, (i,j) \in \mathcal{J})$ such that the vector $\mathbf{s} = (s_{ij}, (i,j) \in \mathcal{J})$, formed from the unique solution s_{ij} to $\text{Link}_{ij}(V_{ij}; w_{ij})$ for each $(i,j) \in \mathcal{J}$, is the same with the solution of minimum cost multi-commodity problem (6), then $\{w_{ij}, (i,j) \in J\}$ is a set of link weights such that all the commodity flow will be forwarded along the shortest paths. Meanwhile, $\mathbf{s}$ solves the optimal traffic engineering problem $\text{TE}(V, \mathcal{G}, \mathbf{c}, \mathbf{D})$.

Remark 1: Wang. [15] shown that for any given set of routes, it is either shortest-path-reproducible or loopy. A theoretical insight behind the result is that an optimal solution for any traffic engineering problem can always be converted to a set of shortest-paths with respect to some link weights [13]. In this paper, we use the convex optimization theory *directly* show that the universal existence of optimal link weights. The most important is that there is a *close form* of the link weights which are *explicitly* determined by the objective and the optimal distribution of the traffic.

III. Utility Function

In this section we investigate a class of (q, β) utility function, which is motivated by the rate control scheme in congestion control [7], as the objective function in (2a). With different parameter settings, a family of specific utility functions can be derived to extract various needs of ISPs. Then we deeply analyze characteristics of the utility maximization problem with the logarithmic utility objective function.

A. (q, β) *Utility Function*

Encouraged by [7], we investigate mathematic features of the rate control scheme in congestion control and propose a similar utility function for the optimal TE problem.

The (q, β) utility function used as a objective function in (2a) is given by

$$V_{ij}(s_{ij}) = \begin{cases} q_{ij} \log s_{ij} & \text{if } \beta = 1 \\ q_{ij}(1-\beta)^{-1} s_{ij}^{1-\beta} & \text{if } \beta \neq 1. \end{cases} \tag{7}$$

Since the piecewise-linear approximation of the $M/M/1$ delay formula proposed by Fortz et al. [12] is a commonly used cost (objective) function, we make a brief comparison between these two objective functions and the results are plotted in Fig. 2. To be fair, here the link capacity is fixed to be 1s and q_{ij} in (7) is thus also set to be 1s.

Based on Theorem 1 and the above utility function, we can derive a family of utility functions with different parameter settings. Now we will illustrate the engineering interpretation of some specific cases.

Case 1: When q_{ij} and β are both set to be 1s, the traffic distribution $\mathbf{f}$ solves the TE problem (2) with $V_{ij}(s_{ij}) = \log s_{ij}$. From Eq. (3b) and (3c), we can get $w_{ij} = \frac{1}{c_{ij} - f_{ij}}$, *i.e.* the average packet delay on link (i,j) is based on the $M/M/1$ queueing model [1], where $f_{ij} = \sum_{t \in \mathcal{D}} f_{ij}^t$. From the discussion above, we have that if path p^* for (s,t) bears positive traffic $y_{p^*} > 0$, then $\sum_{(i,j) \in p^*} \frac{1}{c_{ij} - f_{ij}} \leq \sum_{(i,j) \in p} \frac{1}{c_{ij} - f_{ij}}$ for any other path p for (s,t). The facts above show that the traffic distribution vector $\mathbf{f}$ *not only* minimizes the average packet queueing delay of (s,t) for all $s, t \in \mathcal{N}$, *but also* minimizes the average delay over all the links.

If a network is running with low utilization, then $f_{ij} \ll c_{ij}$, and therefore, the delay $\frac{1}{c_{ij} - f_{ij}} \approx \frac{1}{c_{ij}}$. As such InvCap recommended by Cisco can be suitable.

Case 2: If q_{ij} and β are respectively set to be c_{ij} and 2, then the traffic distribution $\mathbf{f}$ solves (2) with $V_{ij}(s_{ij}) = \frac{-c_{ij}}{c_{ij} - f_{ij}} = -1 - \frac{f_{ij}}{c_{ij} - f_{ij}}$. In this case, we can see that (2) tries to minimize the total average queueing delay by the $M/M/1$ queueing model with respect to optimal link weights $w_{ij} = \frac{c_{ij}}{(c_{ij} - f_{ij})^2}$ for $(i,j) \in \mathcal{J}$.

Case 3: Let d_{ij} be the processing and propagation delay on link (i,j). If q_{ij} and β are respectively set to be d_{ij} and 0, then the traffic distribution $\mathbf{f}$ solves the TE problem (2) with $V_{ij}(s_{ij}) = d_{ij}(c_{ij} - f_{ij}) = d_{ij} c_{ij} - d_{ij} f_{ij}$. In this case, we can see that (2) tries to minimize the total processing and propagation delay, and we have that the optimal link weights $w_{ij} = d_{ij}$ for unsaturated link $(i,j) \in \mathcal{J}$ and $w_{ij} \geq d_{ij}$ for saturated link (i,j). If $d_{ij} = 1$, we have the minimum hop routing.

We use the network topology in Fig. 1 to illustrate these cases. There are four edges with capacities all being 1s. The nonzero demands are 1 for source-destination pair $(1,3)$ and 0.9 for source-destination pair $(3,4)$, respectively. There are two paths for source-destination pairs $(1,3)$, namely 1-3 and 1-2-3. There is a single path for source-destination pair $(3,4)$, i.e., 3-4. Fig.3 (a) and (b) plot the link weights and the link utilization versus parameter β, when q_{ij} is 1, respectively. Detailed numerical results are shown in TABLE I.

Remark 2: Gourdin et al. [20] and Ben-Ameur et al. [21] proposed the numerical studies giving some insight on the impact of using one objective function rather than another, especially the (q,β) utility functions with $\beta=\infty$ and $\beta=2$. In this paper, we investigate the optimal link weights for (q,β) utility function and firstly show the optimal distribution resulted from a general objective function is the same with that from the logarithmic utility function in Theorem 2.

B. Utility versus Logarithmic Utility Function

In this subsection, we will deeply explore characteristics of the utility maximization problem with the logarithmic utility objective function and show that the solution of such utility maximization problem also satisfies the optimal TE problem.

If link (i,j) can choose an amount to pay per unit time, n_{ij}, and receive in return a spare capacity s_{ij} proportionally to n_{ij}, say $s_{ij}=\frac{n_{ij}}{w_{ij}}$, where w_{ij} could be regarded as a charge per unit flow for link (i,j), the utility maximization problem for link (i,j) becomes
$\text{Link}_{ij}(V_{ij};w_{ij})$

$$\begin{array}{ll} \text{maximize} & V_{ij}(\frac{n_{ij}}{w_{ij}})-n_{ij} \\ \text{subject to} & n_{ij}\geq 0. \end{array} \tag{8}$$

Let $\mathbf{n}=(n_{ij},(i,j)\in\mathcal{J}),\mathcal{D}(\mathbf{n})=\{(i,j)\in\mathcal{J}:n_{ij}>0\}$. We define the optimization problem as
$\text{Network}(\mathcal{G},\mathbf{c},\mathbf{D};\mathbf{n})$

$$\begin{array}{ll} \text{maximize}_{\mathbf{f}^t\geq\mathbf{0}} & \sum_{(i,j)\in\mathcal{D}(\mathbf{n})} n_{ij}\log s_{ij} \\ \text{subject to} & \mathbf{c}-\sum_{t\in\mathcal{D}}\mathbf{f}^t=\mathbf{s}\geq\mathbf{0} \\ & \mathbf{B}\mathbf{f}^t=\mathbf{d}^t,\ \forall t\in\mathcal{D}. \end{array} \tag{9}$$

Note that if $n_{ij}=1$ for $(i,j)\in\mathcal{J}$, then the solution to $\text{Network}(\mathcal{G},\mathbf{c},\mathbf{D};\mathbf{n})$ reduces to the traffic allocation in **Case 1** in Section III-A. If $n_{ij},(i,j)\in\mathcal{J}$, are all integers, then the solution to $\text{Network}(\mathcal{G},\mathbf{c},\mathbf{D};\mathbf{n})$ can be constructed as follows.

For each $(i,j)\in\mathcal{J}$, replace the single link (i,j) by n_{ij} identical sub-links, calculate traffic allocation over the resulting $\sum_{(i,j)\in\mathcal{J}}n_{ij}$ traffic, and then provide link (i,j) the aggregate spare capacity allocated to its n_{ij} associated sub-links. The load *per unit charge* is then equivalent to the traffic distribution in **Case 1**.

Say that $\mathbf{s}$ solves $\text{Network}(\mathcal{G},\mathbf{c},\mathbf{D};\mathbf{n})$ if there exists $(\mathbf{f}^t,t\in\mathcal{D})$ such that $(\mathbf{s},\mathbf{f}^t,t\in\mathcal{D})$ solves the optimization problem (9). For the general theory of constrained convex optimization [2], it follows that $(\mathbf{s},\mathbf{f}^t,t\in\mathcal{D})$ solves problem (9) if and only if there exist Lagrangian multiplier vectors $\boldsymbol{\nu}^t$ and $\mathbf{w}$ that satisfy:

$$\begin{array}{rll} \mathbf{c}-\sum_{t\in\mathcal{D}}\mathbf{f}^t=\mathbf{s}, & \mathbf{B}\mathbf{f}^t=\mathbf{d}^t,\ \forall t\in\mathcal{D} & \\ \frac{n_{ij}}{s_{ij}}-w_{ij} & =0,\ \text{if}\ \ (i,j)\in\mathcal{D}(\mathbf{n}) & \\ -w_{ij} & =0,\ \text{if}\ \ (i,j)\notin\mathcal{D}(\mathbf{n}),s_{ij}>0 & \\ -w_{ij} & \leq 0,\ \text{if}\ \ (i,j)\notin\mathcal{D}(\mathbf{n}),s_{ij}=0 & \\ \nu_j^t-\nu_i^t-w_{ij} & =0,\ \text{if}\ \ f_{ij}^t>0 & \\ & \leq 0,\ \text{if}\ \ f_{ij}^t=0. & \end{array}$$

Theorem 2: There exist vectors $\mathbf{n}=(n_{ij},(i,j)\in\mathcal{J}),\mathbf{w}=(w_{ij},(i,j)\in\mathcal{J})$, and $\mathbf{s}=(s_{ij},(i,j)\in\mathcal{J})$ such that

i) $w_{ij}>0$ and $n_{ij}=w_{ij}s_{ij}$, for $(i,j)\in\mathcal{J}$;
ii) n_{ij} solves $\text{Link}_{ij}(V_{ij};w_{ij})$ (8), for $(i,j)\in\mathcal{J}$;
iii) $\mathbf{s}$ solves $\text{Network}(\mathcal{G},\mathbf{c},\mathbf{D};\mathbf{n})$ (9).

Given any such triple $(\mathbf{n},\mathbf{w},\mathbf{s})$, the vectors $\mathbf{n}$ and $\mathbf{s}$ are uniquely determined, and $\mathbf{s}$ solves $\text{TE}(V,\mathcal{G},\mathbf{c},\mathbf{D})$.

Since Theorem 2 is straightforward, here we do not present the detailed proof. It shows that if each link operator is able to choose a charge per unit time prepares to pay, and if the network allocates spare capacities so that the spare capacity per unit charge is equivalent to the traffic distribution in **Case 1**, then a system optimum is achieved when the link operator's choices of charges and the network's choice of allocated spare capacities are in equilibrium.

IV. A New Routing Protocol: SPEF

We are now in a position to design a new routing protocol based on the above theoretical results. In the following, we first present the distributed algorithms to achieve the optimal link weights, also called the first link weight. And then we derive the second link weights from the conceptual framework Network Entropy Maximization [16].

In the Shortest paths Penalizing Exponential Flow-splitting (SPEF), each router can construct the shortest paths for each destination based on the first link weights and independently calculate the traffic split ratio among all equal-cost shortest paths using only the second link wights, which *not only* achieves the optimal traffic engineering *but also* remains the path diversity.

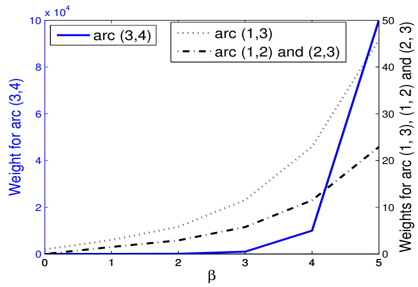
A. Obtaining the First Link Weights

We now show a distributed algorithm to obtain the first link weights, which in fact is the sub-gradient projection method [2] applied to the dual of $\text{TE}(V,\mathcal{G},\mathbf{c},\mathbf{D})$. The algorithm comprises three parts: updating the weight vector, specifying the spare capacity and modifying the routing variables, as described in *Algorithm 1*.

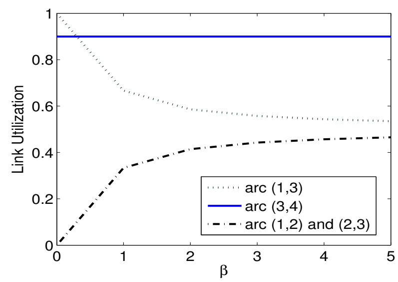
Given the link weight $\mathbf{w}$, the route problem (10) for each destination is a minimum-cost network flow problem [3]. In (11), γ_k is the step size and $(z)_+=\max(0,z)$. And the dual gap

$$\texttt{gap}(\mathbf{w}^{(k)},\mathbf{s}^{(k)},\mathbf{f}^{(k)})=\sum_{(i,j)\in\mathcal{J}}w_{ij}^{(k)}(\sum_{t\in\mathcal{D}}f^{t\,(k)}_{\ ij}+s_{ij}^{(k)}-c_{ij})$$

is selected as the measure of optimality.



(a) Weight vs. parameter β when $q_{ij} = 1$



(b) Link utilization vs. parameter β when $q_{ij} = 1$

Fig. 3. An example illustrating specific TE cases

TABLE I
WEIGHT AND LINK UTILIZATION FOR DIFFERENT OBJECTIVE FUNCTIONS OF TE

Link	$\beta = 0$		$\beta = 1$		B. Fortz & M. Thorup [11]		$\beta \to \infty$		MLU [15]	
	weights	utilizations	weights	utilizations	weights	utilizations	weights	utilizations	weights	utilizations
$(1,3)$	2	1.00	3	0.67	4.6	0.67		0.50	0	$a^{\ddagger}$
$(3,4)$	1	0.90	10	0.90	40.0	0.90		0.90	1	0.90
$(1,2)$	1	0.00	1.5	0.33	2.3	0.33		0.50	0	$1-a$
$(2,3)$	0	0.00	1.5	0.33	2.3	0.33		0.50	0	$1-a$

$^{\ddagger}a$ is a constant in interval $[0.1, 0.9]$

Theorem 3: The link weight sequence $\{\mathbf{w}^{(k)}\}$ generated by Algorithm 1 converges to the first link weights $\mathbf{w}^*$ if $\sum_k \gamma_k = \infty$ and $\gamma_k \to 0$. Furthermore, if there are no saturated links, *i.e.* $s^*_{ij} > 0, \forall (i,j) \in \mathcal{J}$, the first link weights $\mathbf{w}^*$ is uniquely determined and the optimal traffic distribution is $\mathbf{f}^* = \mathbf{c} - \mathbf{s}^*$, where $s^*_{ij} = {V'_{ij}}^{-1}(w^*_{ij})$.

Algorithm 1 *Dual decomposition for the first link weights*

Given tolerance tol and initial weight $\mathbf{w}^{(0)}$ (such as $w^{(0)}_{ij} = 1/c_{ij}), k = 0$;
for the given weight $\mathbf{w}^{(k)}$ **do**
Each link (i,j) solves $\text{Link}_{ij}(V_{ij}; w^{(k)}_{ij})$ to find the spare capacity $s^{(k)}_{ij}$;
Each destination $t \in \mathcal{D}$ solves $\text{Route}_t(\mathbf{w}^{(k)}; \mathbf{d}^t)$:

$$\begin{array}{ll} \text{minimize}_{\mathbf{f}^t \geq \mathbf{0}} & \sum_{(i,j)\in\mathcal{J}} w^{(k)}_{ij} f^t_{ij} \\ \text{subject to} & \mathbf{B}\mathbf{f}^t = \mathbf{d}^t \end{array} \tag{10}$$

to find the routing variable $\mathbf{f}^{t(k)}$;
Each link $(i,j) \in \mathcal{J}$ updates the link weight

$$w^{(k+1)}_{ij} = \big(w^{(k)}_{ij} - \gamma_k (c_{ij} - \sum_{t\in\mathcal{D}} {f^{t}}^{(k)}_{ij} - s^{(k)}_{ij})\big)_+; \tag{11}$$

$k \leftarrow k+1$;
Until $\mathtt{gap}(\mathbf{w}^{(k)}, \mathbf{s}^{(k)}, \mathbf{f}^{(k)}) < tol$.
end for

We have proposed a link weight configuration method that can achieve the optimal traffic engineering. We can determine the set of shortest paths $\text{ON} = \{\text{ON}^t : t \in \mathcal{D}\}$ (*i.e.*, deciding which outgoing link should be chosen on the shortest path) based on the first link weights, where ON^t is the shortest path set for any node $s \in \mathcal{N}$ to destination $t \in \mathcal{D}$. Specifically, SP^r denotes the shortest path set for (s_r, t_r). Let $\text{SP} = \{\text{SP}^r : r \in \mathcal{R}\}$. When the first link weights generate multiple equal cost paths for a source-destination pair or next hops for a given destination routing prefix, we need to split the traffic among the multiple shortest paths or the next hops to keep paths diversity while achieving the optimal traffic engineering.

B. Obtaining the Second Link Weights

Motivated by PEFT [16], we propose an exponential-weighted flow split in the presence of multiple equal cost paths for a given ingress and egress pair (s_r, t_r). The proposed method features that each router can *independently* compute the flow split only based on alternative link weights, where routers can direct traffic on the shortest paths determined by the first link weights. This method can achieve network-wide traffic engineering objective through OSPF while keeping the simplicity and scalability of link-state routing protocols.

We maximize the relative entropy of the traffic split vector among the multipath in SP^r to maintain the path diversity. Maximizing the relative entropy [**?**] of the traffic split vector can be formulated as follows.
$\text{NEM}(\text{SP}, \mathbf{f}^*, \mathbf{D})$:

$$\text{maximize} - \sum_{r\in\mathcal{R}} d_r \sum_{k=1}^{n_r} p^r_k \log p^r_k \tag{12a}$$

$$\text{subject to} \sum_{r\in\mathcal{R}} \sum_{k:(i,j)\in\text{SP}^r_k} d_r p^r_k \leq f^*_{ij}, \forall (i,j) \in \mathcal{J} \tag{12b}$$

$$\sum_{k=1}^{n_r} p^r_k = 1, \ \forall r \in \mathcal{R}, \tag{12c}$$

where n_r denotes the number of the shortest paths from s_r to t_r. SP^r_k denotes the k-th shortest path from s_r to t_r.

Remark 3: Comparison the NEM problem proposed for PEFT, the $\mathrm{NEM}(\mathrm{SP}, \mathbf{f}^*, \mathbf{D})$ only splits the traffic on the shortest paths determined by the first link weights. The resulted benefit is we can design Algorithm 3 obtaining the optimal solution, but not a heuristic algorithm as in [16] for PEFT.

We will show that the optimal solution to (12) is realizable with hop-by-hop forwarding to exponential penalty. Let $(\mathbf{p}^r, r \in \mathcal{R})$ be a solution of (12). Then there exist Lagrangian multipliers vector $\mathbf{v} = (v_{ij}, (i,j) \in \mathcal{J})$ and $(\nu_r, r \in \mathcal{R})$ satisfying that $(1 + \log p_k^r) + \sum_{(i,j)\in \mathrm{SP}_k^r} v_{ij} + \frac{\nu_r}{d_r} = 0,\ \forall r \in \mathcal{R}, k$ and $\sum_{k=1}^{n_r} p_k^r = 1$. Under these conditions, we have

$$p_k^r = \frac{e^{-v_k^r}}{\sum_{i=1}^{n_r} e^{-v_i^r}},\ \forall r \in \mathcal{R}, k, \tag{13}$$

where $v_k^r = \sum_{(i,j)\in \mathrm{SP}_k^r} v_{ij}$. In the following, we refer to Lagrange multipliers vector $\mathbf{v}$ as the second link weight. v_k^r is the length of path SP_k^r with respect to the second weight $\mathbf{v}$.

Theorem 4: The optimal traffic engineering for a given traffic can be realized with the second link weights using exponential flow split (13).

To provide a foundation for the second link weight computation, we investigate the Lagrange dual problem of $\mathrm{NEM}(\mathrm{SP}, \mathbf{f}^*, \mathbf{D})$ and a dual-gradient-based solution. Denote the dual variables for constraints (12b) as v_{ij} for link (i,j) (or $\mathbf{v}$ as a vector). We first write the Lagrangian $L(\mathbf{p}, \mathbf{v})$ associated with problem $\mathrm{NEM}(\mathrm{SP}, \mathbf{f}^*, \mathbf{D})$ as

$$L(\mathbf{p}, \mathbf{v}) = -\sum_{r\in\mathcal{R}} d_r \sum_{k=1}^{n_r} (p_k^r \log p_k^r + v_k^r p_k^r) + \sum_{(i,j)\in\mathcal{J}} v_{ij} f_{ij}^*,$$

where $v_k^r = \sum_{(i,j)\in \mathrm{SP}_k^r} v_{ij}$. The Lagrange dual function is

$$\begin{aligned} d(\mathbf{v}) = \text{maximize} \quad & L(\mathbf{p}, \mathbf{v}) \\ \text{subject to} \quad & \textstyle\sum_{k=1}^{n_r} p_k^r = 1,\ \forall r \in \mathcal{R}. \end{aligned}$$

The dual problem is then formulated as

$$\text{minimize } d(\mathbf{v}) \text{ subject to } \mathbf{v} \geq \mathbf{0}. \tag{14}$$

To solve the dual problem, we first consider the maximization of the Lagrangian over $\mathbf{p}$. Note that, the $L(\mathbf{p}, \mathbf{v})$ is separable for a given dual variable $\mathbf{v}$, *i.e.*, the traffic split subproblem for each $r \in \mathcal{R}$ is independent of the others since they are not coupled together with link capacity constraint (12b). So we can solve a subproblem (15) below for each $r \in \mathcal{R}$ separately:

$$\begin{aligned} \text{maximize} \quad & -d_r \textstyle\sum_{k=1}^{n_r} \left(p_k^r \log p_k^r + v_k^r p_k^r\right) \\ \text{subject to} \quad & \textstyle\sum_{k=1}^{n_r} p_k^r = 1. \end{aligned} \tag{15}$$

Then, the dual problem (14) can be solved by using the gradient projection method as follows for iterations indexed by k,

$$\begin{aligned} v_{ij}^{(k+1)} &= \left(v_{ij}^{(k)} - \gamma(f_{ij}^* - \textstyle\sum_{r\in\mathcal{R}} d_r \sum_{l:(i,j)\in \mathrm{SP}_l^r} p_l^{r\,(k)})\right)_+ \\ &= \left(v_{ij}^{(k)} - \gamma(f_{ij}^* - f_{ij}^{(k)})\right)_+ \end{aligned} \tag{16}$$

TABLE II

FORWARDING TABLE FOR SPEF ROUTING.

Next hop	Lengths of multiple equal cost shortest paths through link $(s, \text{next hop})$ to t in view of the second link weights
v_1	$(v_{11}^{(s,t)}, \cdots, v_{1n_1}^{(s,t)})$
$\vdots$	$\vdots$
v_{m_s}	$(v_{m_s 1}^{(s,t)}, \cdots, v_{m_s n_{m_s}}^{(s,t)})$

where $\gamma > 0$ is a constant step size, $(p_1^{r\,(k)}, \cdots, p_{n_r}^{r\,(k)})$ are solutions of the traffic split subproblem (15) for $\mathbf{v}^{(k)}$, and $f_{ij}^{(k)}$ is the total flow on link $(i,j) \in \mathcal{J}$.

It is important to note, from (16) in iteration $k+1$, the procedure of link weight updating needs $f_{ij}^{(k)}$, the aggregate bandwidth usage. We now show how to calculate it efficiently.

First, we need to establish the forward table for node s to destination t as shown in Table II, where n_k denotes the number of shortest path from node s through node v_k to node t, $v_{kj}^{(s,t)}$ is the length of the j-th path from node s through node v_k to node t, and m_s denotes the number of next hop for s in ON^t. Then the traffic to destination t can be split according to the formula

$$\Gamma^t(s, v_k) = \frac{\sum_{j=1}^{n_k} e^{-v_{kj}^{(s,t)}}}{\sum_{i=1}^{m_s} \sum_{j=1}^{n_i} e^{-v_{ij}^{(s,t)}}},\ k = 1, \cdots, m_s. \tag{17}$$

Finally, the formal algorithm for the second link weights can be described as follows, in which Algorithm 3 is needed to get the traffic distribution matching to the current second link weights $\mathbf{v}^{(k)}$.

Algorithm 2 *Dual decomposition for the second link weights*

Input the optimal traffic distribution $\mathbf{f}^*$ and tolerance ϵ;
Given the initial second link weights $\mathbf{v}^{(0)} = \mathbf{0}, k = 0$;
For the given weights $\mathbf{v}^{(k)}$, **do**
  Get the traffic distribution matching to $\mathbf{v}^{(k)}$, i.e.

$$\mathbf{f}^{(k)} \leftarrow \mathrm{TrafficDistribution}(\mathbf{v}^{(k)}).$$

  Each link (i,j) updates the second link weights

$$v_{ij}^{(k+1)} \leftarrow \left(v_{ij}^{(k)} - \gamma(f_{ij}^* - f_{ij}^{(k)})\right)_+;$$

  $k \leftarrow k+1$;
Until $f_{ij}^{(k)} \leq f_{ij}^* + \epsilon$ for all $(i,j) \in \mathcal{J}$.

The following result can be proved with standard convergence analysis for gradient projection algorithms [2]:

Theorem 5: Let $\{\mathbf{v}^{(k)}\}$ be the sequence generated by Algorithm 2. We have that $\{\mathbf{v}^{(k)}\}$ converges to the optimal dual solutions $\mathbf{v}^*$, and the corresponding primal variables $\mathbf{p}^*$ according to (13) is the globally optimal solution of (12).

Algorithm 3 *TrafficDistribution*($\mathbf{v}$)

Input $\mathrm{ON} = \{\mathrm{ON}^t : t \in \mathcal{D}\}$;
Compute the path length for each path in ON
  in view of the second link weights $\mathbf{v}$;
Compute the traffic split $\Gamma^t(i,j)$ according to (17);
For each destination t **do**
  Do sorting on the distance of node s to t
   in view of the first link weights
  Each source $s \neq t$ in the decreasing distance order **do**
   $\bar{d}_{st} = d_{st} + \sum_{(j,s)\in \mathrm{ON}^t} f^t_{js}$;
  For all j such that $(s,j) \in \mathrm{ON}^t$
   $f^t_{sj} = \bar{d}_{st}\Gamma^t(s,j)$;
  end for
end for
$f_{ij} = \sum_{t\in\mathcal{D}} f^t_{ij}$ for all $(i,j) \in \mathcal{J}$;
Return /* set of **f** */

Here $\bar{d}_{st}$ denotes the total incoming flow destined to node t at node s (including traffic originating at s as well as any traffic arrived from other nodes).

We now present a new link-state routing with hop-by-hop forwarding, which can achieve the optimal traffic engineering.

Algorithm 4 *SPEF routing*

Running Algorithm 1 to obtain the first link weights $(w_{ij}, (i,j) \in \mathcal{J})$ and optimal traffic distribution $\mathbf{f}^*$.
For each destination node $t \in \mathcal{D}$ **do**
  Run Dijkstra's algorithm with the first link weights
   to get all the shortest paths $\mathrm{ON} = \{\mathrm{ON}^t : t \in \mathcal{D}\}$.
end **for**.
Running Algorithm 2 to obtain the second link weights $(v_{ij}, (i,j) \in \mathcal{J})$.
For each $t \in \mathcal{D}$ **do**
  For each source node s:
   Establish the forward routing table shown in Table II.
  end For
end For

V. Performance Evaluation

How well can the new routing protocol SPEF perform? In the first part, we will illustrate its performance with a simple example. In the second part, we demonstrate the performance of SPEF with numerical experiments over two real backbone networks and several synthetic networks. Here we make comparison between SPEF and OSPF, where the latter protocol sets link weight inversely proportional to its capacity and evenly splits the traffic on the set of equal-cost next hops.

A. An Example

Fig. 4 shows a simple network topology, as used in [15]. Each link has a capacity of 5 units and each demand needs a bandwidth of 4 units. For simplicity, we omit six links unused. The numbers on the links are the link indices.

The link utilizations for optimal TE with a different parameter β are shown in Fig. 6. For the results of $\beta = 0$, link 1 is a bottle link. And the first link weight is 3. The first link weight of others are all 1s. Considering link 1, the link utilization is decreasing in β. From Eq. (3b), the first weight of links 2

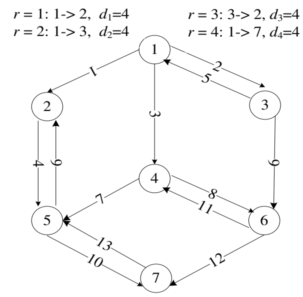


Fig. 4. A simple network topology and traffic demands

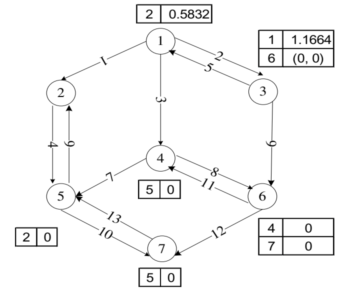


Fig. 5. SPEF forwarding table for destination 2

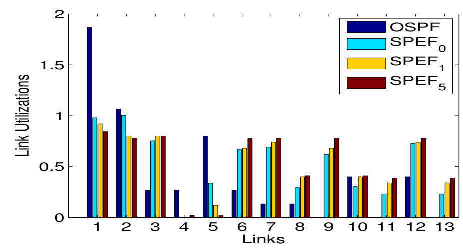


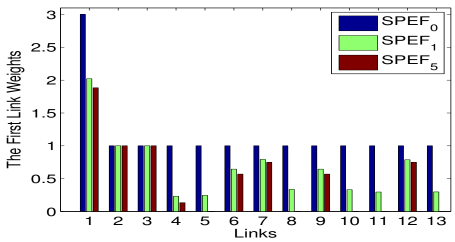
Fig. 6. The link utilization for the topology shown in Fig.4

and 3 are the same when $\beta = 0, 1$ or 5, since all the spare capacities are equal to 1. For $\beta = 1$, from Fig.7 (b), it can be seen that all the second link weights are zero except for link 1 and link 5. The fact that the second weight of link 1 is increasing in β shows we route fewer traffic through link 1 with larger β.

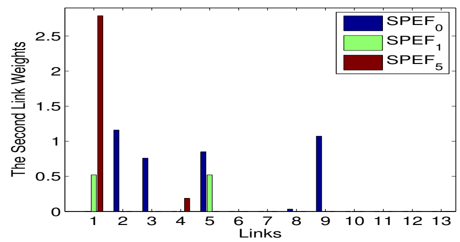
B. Simulation Environment

The properties of the networks used are summarized in TABLE III. The real backbone networks, the Abilene network and Cernet2 network shown in Fig.8. The first network has 11 nodes and 28 directional links with 10Gbps capacity, and the latter has 20 nodes and 44 directional links with 10Gbps capacity for 8 backbone links (plotted by bold black) and 2.5Gbps for others. The traffic demands for Abilene network is generated as those in Fortz and Thorup [12]. The traffic demands for Cernet2 network are generated by a gravity model with the link aggregated load extracted from the sample Netflow data, which was captured during 2010/1/10 to 2010/1/16. To simulate networks with different congestion levels, we create different test cases by uniformly increasing the traffic demands until the maximal link utilization almost reaches 100% with SPEF.

We test the algorithms proposed in this paper on the same topologies and traffic matrices as in Fortz and Thorup [12]. The 2-level hierarchical networks were generated using GT-ITM, which consist of two kinds of links: local access links with 1 unit capacity and long distance links with 5-unit capacity. In the random topologies, the probability of having a link between two nodes is a constant parameter, and all link capacities are 1 unit. In these test cases, for

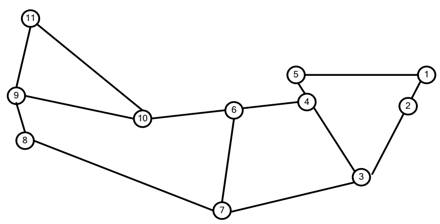


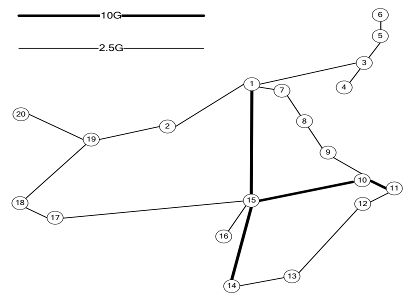
(a) The first link weights



(b) The second link weights

Fig. 7. The first and second link weights for the network topology shown in Fig. 4 with different β


(a) Abilene network



(b) Cernet2 network

Fig. 8. Backbone network topologies

TABLE III
PROPERTIES FOR DIFFERENT NETWORKS

Net. ID	Topology	Node #	Link #
Abilene	Backbone	11	28
Cernet2	Backbone	20	44
Hier50a	2-level	50	222
Hier50b	2-level	50	152
Rand50a	Random	50	242
Rand50b	Random	50	230
Rand100	Random	100	392

each network, traffic demands are proportionally increased to simulate different congestion levels.

For SPEF, we employ the utility function with $\beta = 1$ to determine the first link weights. The resulted utility is normalized, which means $\sum_{(i,j)\in\mathcal{J}} \log(1-u_{ij})$, where u_{ij} is the link (i,j)'s utilization. The utility is $-\infty$ if MLU is greater than 1, which is not shown in Fig. 10.

C. Performance Comparison Against OSPF

The sorted link utilizations for Abilene network and Cernet2 network are shown in Fig. 9, where the network load is the ratio of total demand over the total capacity. Typical results for different topologies are shown in Fig.10.

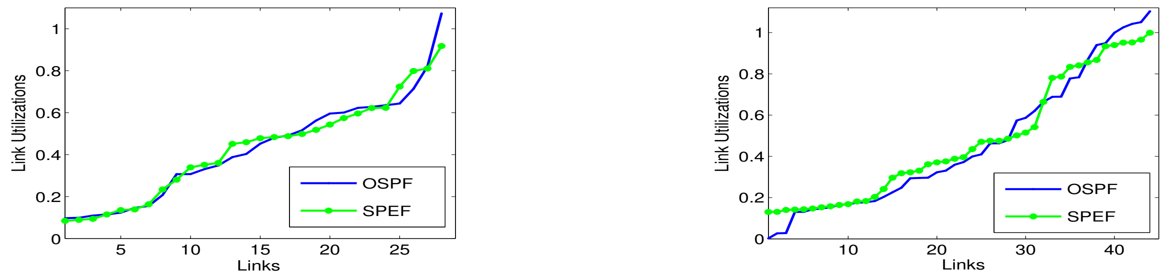
From Fig. 9, it can be seen that some underutilized links in OSPF are used efficiently in SPEF. At the same time the traffic on the over-utilized links in OSPF is removed in SPEF. The results shown in Fig. 10 indicate that the utility difference between SPEF and OSPF becomes obvious with the increasing of network load. SPEF still works when MLU of OSPF is greater than 1.

Due to space limitation, more comparison results between SPEF and PEFT could be found in our extended version [22].

D. Convergence Behavior

In Algorithm 1, the initial link weights $w_{ij}^{(0)} = \frac{1}{c_{ij}}$ for all link $(i,j) \in \mathcal{J}$ are a proper choose. The step sizes in Algorithm 1 can be constant or dynamically adjusted. We find that setting the step size in Algorithm 1 to the reciprocal of the maximum link capacity $\frac{1}{\max\{c_{ij}:(i,j)\in\mathcal{J}\}}$ performs well in practice. Fig. 11 (a) shows the evolution of dual objective value of TE obtained by Algorithm 1 with different step sizes, within the first 2000 iterations for Cernet2 network. It provides convergence behavior typically observed. The legends show the ratio of the step size over the default setting which is $\frac{1}{\max\{c_{ij}:(i,j)\in\mathcal{J}\}}$. It demonstrates that Algorithm 1 developed for the SPEF routing convergence very fast with default setting.

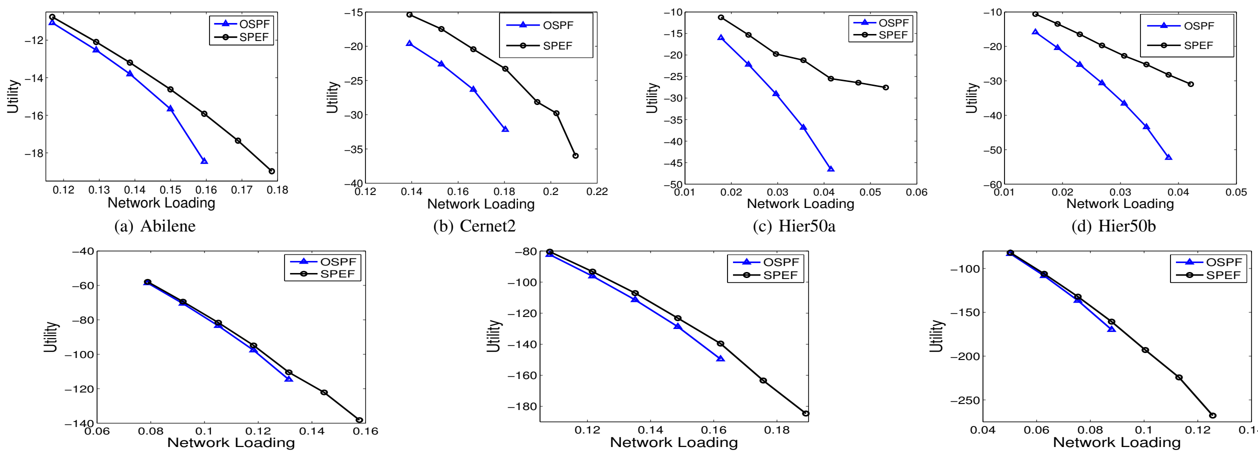
In Algorithm 2, the initial link weights $v_{ij}^{(0)} = 0$ for all link $(i,j) \in \mathcal{J}$ are a proper choose. We find that setting the step size in Algorithm 2 to the reciprocal of the maximum optimal traffic distribution $\frac{1}{\max\{f_{ij}^{*}:(i,j)\in\mathcal{J}\}}$ performs well in practice. Fig. 11 (b) shows evolution of dual objective value of NEM obtained by Algorithm 2 with different step sizes for Cernet2 network. It provides convergence behavior typically observed. The legends show the ratio of the step size over the default setting which is $\frac{1}{\max\{f_{ij}^{*}:(i,j)\in\mathcal{J}\}}$. It demonstrates that the initial link weights for Algorithm 2 are a good approximation solution for the dual problem of NEM. And Algorithm 2 developed for the SPEF routing also convergence very fast with default setting. Algorithm reduces the dual objective value of NEM to 0.6695 after 100 iterations and



(a) Abilene with network load 0.17

(b) Cernet2 with network load 0.21

Fig. 9. Comparison of SPEF and OSPF in terms of the sorted link utilization



(a) Abilene (b) Cernet2 (c) Hier50a (d) Hier50b

(e) Rand50a (f) Rand50b (g) Rand100

Fig. 10. Comparison of SPEF and OSPF in terms of utility

0.66945 after 300 iterations. In addition, increasing step size a little will speed up the convergency.

E. Equal Cost Paths

One of the key features of SPEF routing is the ability to balance traffic across multiple equal-cost paths. Intuitively, SPEF routing is more likely to use multiple paths to balance traffic at higher loads. Hence, we focus on a different utilization scenarios for Cernet2 network, for which we compute the number of equal cost paths used by SPEF routing. TABLE IV shows the results, where n_i denotes the number of ingress-egress pairs that have i equal cost paths. It can be seen that the equal cost paths for some ingress-egress pairs are increasing with the increase of network load. But OSPF routing has not change with the network load.

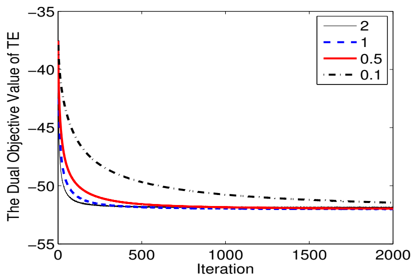
TABLE IV
COMPARISON OF SPEF AND OSPF IN TERMS OF THE NUMBER OF EQUAL COST PATH FOR EACH INGRESS-EGRESS PAIR

Routing	Network loading	n_1	n_2	n_3	n_4
OSPF	0.13, 0.17, 0.21	355	25	0	0
SPEF	0.13	330	48	0	2
	0.17	325	53	0	2
	0.21	321	54	3	2

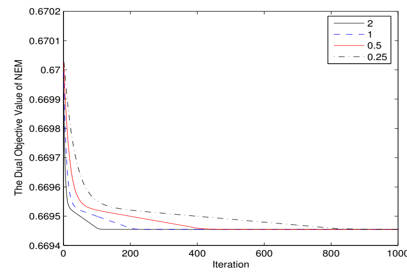
VI. RELATED WORK

Among the papers focused on TE, MLU [15] and piecewise-linear approximation of the $M/M/1$ delay formula [11] are two frequently-used cost functions. Minimizing MLU ensures that the traffic is moved away from congested hot spots to less utilized parts of the network. The latter formula proposed by Fortz et al. [11] is based on discussions with the technicians in AT&T Lab. Srivastava et al. [14] constructed a composite cost function which was a positive linear combination of the used capacity and MLU, and then proposed a heuristic hybrid method combining the sub-gradient projected method and a genetic algorithm to determine the link weight system. Yuan [19] proposed an approach for robust OSPF routing using an artificial objective function embedded into a local search algorithm.

Researchers in group of congestion control are mainly concerned with fairness and efficiency. Network utility maximization (NUM) [4], especially the proportionally fair, is a trade-off objective for this aim. In addition, to design the end-to-end algorithms for joint routing and rate control, many following researches replace capacity constraints with barrier functions that specify the congestion cost at the link (e.g., [4],



(a) Evolution of dual objective value of Algorithm 1



(b) Evolution of dual objective value of Algorithm 2

Fig. 11. Evolution of dual objective value obtained by Algorithm 1 and Algorithm 2 with different step sizes for Cernet2 network

[6]). Generally, a function $\Phi_{ij}(f_{ij})$ is defined, which can be regarded as a penalty function that describes the rate at which the cost is incurred at resource (i, j) with capacity c_{ij} when the load through it is f_{ij}. He et al. [8] choose $\Phi(f_{ij}, c_{ij}) = e^{\frac{f_{ij}}{c_{ij}}}$ to model $M/M/1$ queuing delay. Xu et al. in [17] defined $\Phi_{ij}(f_{ij}) = -q_{ij}\ln(c_{ij} - f_{ij})$.

Fortz et al. [12] showed that optimizing the link weights for OSPF with evenly split over ECMP to the offered traffic is an NP-hard problem and proposed a local search heuristic. Sridharan et al. [13] used a centralized greedy computation to select the subset of next-hops for each prefix to attain load balance much better than even splitting among the shortest paths. But these solutions fail to enable routers to independently compute the flow-splitting ratios only using link weights. PEFT, recently proposed by Xu et al. [16], is a promising link-state routing protocol splitting traffic over multiple paths with an exponential penalty on longer paths. In order to prevent loops and promote computational efficiency, PEFT used *Downward PEFT* for traffic splitting, which does not provably achieve optimal TE [16]. However, PEFT shed a new light for studies on developing an OSPF-based protocol.

VII. Conclusion

In this paper, we explore the problem of achieving the optimal traffic engineering in intra-domain IP networks. We model the optimal TE as the utility maximization of multi-commodity flows and theoretically show that any given set of optimal routes corresponding to a particular objective function can be converted to shortest paths with respect to a set of positive link weights, which can be directly configured on OSPF-based protocols. On these bases, we develop a new OSPF-based routing protocol, SPEF, to realize a flexible way that splits traffic over shortest paths in a distributed fashion. The inspiring fact lies that comparing to OSPF, SPEF only needs one more weight for each link and provably achieves optimal TE. Numerical experiments have been done to compare SPEF with the current version of OSPF, showing the effectiveness of SPEF in terms of link utilization and network traffic distribution.

A direction for further studies is that we should analyze the computational complexity in network environment with OSPF as well as other existing approaches including PEFT.

References

[1] D. P. Bertsekas and R. Gallager. *Data Networks*, Englewood Cliffs, NJ: Prentice-Hall, 1992.

[2] D. P. Bertsekas. *Nonlinear Programming*, 2nd ed. Belmont, MA, USA: Athena Scientific, 1999.

[3] J. Z. Zhang and S. J. Xu. *Linear programming*, Beijing, P.R. China:Science Publishing House, 1990.

[4] F. Kelly, A. Maulloo and D. Tan. "Rate control for communication networks: shadow prices, proportional fairness and stability", *J. Operations Res. Soc.*, vol. 49, No.3, pp.237-252, 1998.

[5] F. P. Kelly. "Charging and rate control for elastic traffic (with correction)", *Eur Trans. on Telecommun*, vol.8, pp.33-37, 1997.

[6] F. P. Kelly and T. Voice. "Stability of end-to-end algorithms for joint routing and rate control", *Computer Communication review*, vol. 35, no. 2, pp. 5-12, 2004.

[7] J. Mo, J. Walrand. "Fair end-to-end window-based congestion control", *IEEE/ACM Trans. on Networking*, vol.8, no.5, pp.556-567, Oct. 2000.

[8] J.He, M. Suchara and M. Chiang. "Rethinking Internet traffic management: from multiple decompositions to a practical protocol", In *CoNEXT'07*, New York:2007.

[9] D. Awduche. "MPLS and traffic engineering in IP networks", *IEEE Communication Magazine*, vol. 37, no. 12, pp. 42-47, Dec. 1999.

[10] Cisco, Configuring OSPF, 1997.

[11] B. Fortz and M. Thorup. "Internet traffic engineering by optimizing OSPF weights", *Proc. IEEE INFOCOM, 2000*, Tel Aviv, Israel, 2000.

[12] B. Fortz and M. Thorup. "Increasing Internet Capacity Using Local Search", *Computational Optimization and Applications*, vol. 29, pp. 13-48, 2004.

[13] A. Sridharan, R. Guerin and C. Diot. "Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks", *IEEE/ACM Trans. on Networking*, vol.13, No. 2, pp. 234-247, Apr. 2005.

[14] Srivastava, S., Agrawal, G., Pioro, M., and Medhi, D. 2005. Determining link weight system under various objectives for OSPF networks using a Lagrangian relaxation-based approach. *IEEE e-Trans. on Network and Service Management*, vol. 2, No.1, pp.9-18.

[15] Z. Wang, Y. Wang and L. Zhang. "Internet traffic engineering without full mesh overlaying", *Proc. IEEE INFOCOM, 2001*, Anchorage, AK.

[16] D. Xu, M. Chiang, and J. Rexford. "Link-state routing with hop-by-hop forwarding achieves optimal traffic engineering", *Proc. IEEE INFOCOM, 2008*, Phoenix, USA.

[17] K. Xu, H. Liu, J. Liu and J. Zhang. "LBMP: A Logarithm-Barrier-based Multipath Protocol for Internet Traffic Management", *IEEE Trans. on Parallel and Distributed Systems*, vol.22, no.3, pp.456-470, 2011.

[18] H. Wang, H. Xie and L. Qiu. "COPE: traffic engineering in dynamic networks". *Proc. SIGCOMM'06*, Sep. 11-15, 2006, Pisa, Italy.

[19] D. Yuan. "A Bi-Criteria Optimization Approach for Robust OSPF Routing". *Proc. 6th IEEE Workshop on IP Operation Management (IPOM 2003)*, 2003.

[20] W. Ben-Ameur, É. Gourdin, B. Liau and N. Michel. "Routing strategies for IP networks", *Telektronikk Magazine*, vol. no. 2/3, pp. 145-158, 2001.

[21] É. Gourdin, and O. Klopfenstein. "Comparison of different QoS-oriented objectives for multicommodity flow routing optimization", *Proc. 13th International Conference on Telecommunications (ICT 2006)*, Funchal, Portugal, 2006.

[22] http://arxiv.org/abs/1011.5015/.

[23] J. Moy. RFC 2328 OSPF Version 2, http://tools.ietf.org/html/rfc2328.