TROPIC: Traffic-Engineering-Oriented Planning of IP Core Networks

Leon Richardt°, Alexander Brundiers•°, Timmy Schüller•, Nils Aschenbruck°

°Osnabrück University, Institute of Computer Science
Osnabrück, Germany
{richardt, brundiers, aschenbruck}@uos.de

•Deutsche Telekom Technik GmbH
Münster, Germany
{alexander.brundiers, timmy.schueller}@telekom.de

***Abstract*—Network expansion and traffic engineering (TE) are the two prevalent approaches for network operators to deal with the growth of Internet traffic. Although these are two complementary approaches, they are generally conducted completely independent of each other. In this paper, we argue for the importance of integrating routing and TE schemes already during expansion planning. For this, we introduce the *2SR Network Expansion Problem* (2SR-NEP) that, given a network topology and a traffic forecast, aims for a minimum-cost network expansion that satisfies the projected demands under the 2-Segment Routing paradigm. We prove the NP-hardness of the 2SR-NEP and propose the TROPIC algorithm to solve it heuristically. An evaluation on a publicly available dataset shows the approach's ability to achieve cost savings of up to 60% over a baseline approach.**

I. Introduction

The volume of IP traffic which Internet Service Providers (ISPs) must handle is projected to keep growing at significant rates, largely driven by multimedia applications [1], [2]. In order to uphold service-level agreements with customers, network operators must ensure enough capacity is available to deliver every traffic demand. Therefore, the task of planning and evolving link capacities is of critical importance for ISPs. Since long-term capacity expansion typically requires significant capital investment, it is crucial to find cost-effective solutions. Aside from expanding link capacities, ISPs today make use of traffic engineering (TE) to optimize the load distribution in the network. Traditionally, however, the joint consideration of network planning and TE has been unrealistic, assuming routing schemes that are either significantly less or significantly more powerful than what current technology permits. In fact, prominent networking hardware vendors such as Cisco [3] or Juniper [4] also segregate their tool suites for capacity planning and TE into separate components.

In this paper, we show that an integrated planning approach using 2-Segment Routing (2SR) can provide significant cost savings compared to a baseline approach that reflects planning behavior still commonly used today. In order to make the setting amenable to theoretical analysis, we propose the 2SR Network Expansion Problem (2SR-NEP) to model the problem. Specifically, we make the following contributions:

- We introduce the 2SR-NEP and prove its NP-hardness.
- We propose the heuristic TROPIC algorithm to solve the 2SR-NEP in acceptable time even for large networks.
- We provide a comprehensive evaluation of TROPIC on instances derived from the REPETITA dataset [5], achieving cost reductions of up to 60% compared to a greedy baseline approach.

In Sec. II, we introduce relevant background knowledge and discuss related work. In Sec. III, we introduce our network model in order to formally define the 2SR-NEP and prove its NP-hardness. Sec. IV proceeds to describe the implementation of the TROPIC algorithm. In Sec. V, we describe the synthetic dataset used in the evaluation. The evaluation approach, its results, and its limitations are discussed in Sec. VI before the paper is concluded with a recapitulation of our most important findings and a discussion of possible future work in Sec. VII.

II. Background

As the 2SR-NEP concerns the domains of Segment Routing for Traffic Engineering (SR-TE) and capacity planning, this section introduces relevant literature for both. At the end of the section, we discuss the limitations of existing approaches and the potential benefits of an integrated planning approach.

A. Segment Routing

Segment Routing [6] is a network architecture following the source-routing paradigm. An SR-enabled router can choose to apply a list of segments (a specific node, adjacency, or service) to any packet it injects into the network. The packet is routed between the individual waypoints given by the segments along the shortest paths defined by the Interior Gateway Protocol (IGP). Due to its immaculate traffic steering capabilities combined with an exceptionally low overhead, SR has become one of, if not the premier technological choices for many network operators, resulting in lots of attention from both industry [7]–[9] and academic research [10]. Although a multitude of TE objectives are considered in the academic literature, a major focus lies on the minimization of the network's maximum link utilization (MLU). There are different approaches towards solving the SR-TE problem, e.g., based on linear programming (LP) [11], column generation [12], constraint programming [13], or meta-heuristic approaches [14].

Although there is some literature that studies SR's performance under short-term changes in the topology such as node or link failures [15]–[17], we are not aware of any study that considers long-term expansions of the network through capacity upgrades or link additions.

ISBN 978-3-903176-72-0 © 2025 IFIP

B. Capacity Planning

At some point, logical changes to routing paths are no longer sufficient to handle a growing traffic volume, and physical network expansion becomes necessary. There are many ways to pose the network design problem, see, e.g., [18] for an overview. However, as pointed out in [19], "little has been revealed about production network planning" of large network operators such as ISPs or hyperscalers. In [20], Holmberg and Yuan present a generalized LP formulation for different network design problems. They show that the optimization of link weights can significantly reduce planning costs compared to the default weight-setting strategy employed in OSPF, even when the network flows are only considered sequentially.

Gerstel *et al.* use multi-layer planning to reduce overprovisioning in IP-optical networks [21]. Their method relies on modelling hardware failures at the optical layer and resolving them using an integrated IP-optical restoration approach. They do not proactively adjust routing paths. Gkamas *et al.* also present a multi-layer planning algorithm [22]. Their approach considers routing implicitly by assigning IP/MPLS links to light paths, but it does not allow the use of TE when assigning routes. Other approaches (e.g. [23]) use a general path model with binary decision variables for each possible routing path. This typically runs into scaling issues for networks of larger sizes. Additionally, such models assume arbitrarily expressive traffic steering capabilities, which cannot realistically be implemented in practice.

Ahuja *et al.* present the planning model used for Facebook's backbone network which relies on yet a different routing scheme [19]. Here, the flows are routed according to a fractional multi-commodity flow (MCF) formulation. The authors explain that they account for this overly optimistic assumption by multiplying the traffic demands with a routing overhead parameter. However, they neither go into detail about how the *actual* routing policies are obtained from the MCF result when a planning solution is to be implemented, nor do they offer any guidance on how to choose the overhead parameter.

C. Benefits of Integrated Planning

While the above weight-tuning and MCF-based planning paradigms may be able to achieve reasonably good theoretical results, there are several issues when it comes to a practical deployment of the respective solutions, which can be resolved by the SR-based, integrated approach proposed in this paper.

IGP weight tuning is inflexible. While metric-based TE can often achieve near-optimal results in regard to link utilization [24], it lacks the traffic steering flexibility available with modern TE techniques such as SR. In particular, with SR-TE, traffic can be steered on a *per-flow basis*, allowing network operators to use different paths for different traffic flows between the same origin-destination pair, tailored to the requirements of a specific traffic flow (e.g., guaranteeing low latency for time-sensitive data, or avoiding specific nodes for data privacy). Weight tuning does not offer such flexibility. Furthermore, tuned link weights may no longer follow any real-world property (e.g., capacity or latency), making them difficult to understand even for experts. In practice, network operators prefer a routing scheme that allows a "human in the loop" to comprehend the installed paths and the effects that would result from changing them [9].

Thus, although weight tuning is still used in present-day networks, many ISPs are looking to migrate to SR as the dominant TE technique. Consequently, the planning of future network expansions should be based on the SR-TE paradigm.

MCF is not realistic. The category of weight-tuning planning approaches assume less than what is possible with current technologies. In contrast, the category of MCF-based planning approaches assumes too much, as arbitrary routing paths cannot realistically be implemented. Thus, operators are required to translate the routes into a practical routing scheme. This may lead to significant degradation of the solution quality. By planning with SR from the beginning, operators can be sure that the planning result corresponds closely to the real network conditions and constraints.

III. Problem Definition

This section describes assumptions and requirements regarding the network and forecasting model on which our work will be based. Building on this, we formally define the 2SR-NEP and prove its NP-hardness.

A. Network and Forecasting Model

The IP network under study is given as a graph $G = (V, A)$ with bidirected edges (links). Each link $l \in A$ is attributed with a capacity $c(l) \in \mathbb{Q}_{\geq 0}$ and a routing weight $w(l) \in \mathbb{Q}_{+}$. It holds that $(u, v) \in A$ for each $(v, u) \in A$ and, more strictly, $c(l) = c(r(l))$.[1] We assume that IP flows in the network can be routed with 2SR.

The set of links that should be considered in the expansion planning process (the *candidate links*, CLs) is given by $K = B \uplus I$ wherein $B \subseteq A$ describes the *existing* links to be expanded and $I \subseteq V^2 \setminus A$ are the *innovative* candidate links (ICLs) that can be newly added to the network. In order to preserve the bidirectionality of links in a planning solution, we require that $(u, v) \in K$ for each $(v, u) \in K$. For each CL $l \in B$, there is a modular capacity $\mu(l) \in \mathbb{Q}_{+}$ as well as an expansion price $\xi(l) \in \mathbb{Q}_{\geq 0}$ defining the cost of adding $\mu(l)$ units of capacity to l. An ICL $i \in I$ is also equipped with an initial capacity $\eta(i) \in \mathbb{Q}_{+}$ and an associated addition cost $\zeta(i) \in \mathbb{Q}_{\geq 0}$, representing the upfront investment required to add i to the network. Before an innovative link can be expanded it must be added to the network. All costs in our model can be considered as total costs of ownership (TCO).

We assume a forecast of the traffic demands at the planning horizon is available in the form of a traffic matrix (TM) $\mathbf{T} = (t_{u,v})_{u,v \in V}$ where $t_{u,v} \in \mathbb{Q}_{\geq 0}$ represents the traffic demand from node u to node v.

[1]For every link l, we refer to the link in the reverse direction as $r(l)$.

B. The 2SR Network Expansion Problem (2SR-NEP)

Let $U_T \in \mathbb{Q}_+$ give the acceptable threshold for the MLU when $\mathbf{T}$ is routed in the expanded network. This parameter is chosen by network operators as a tradeoff between efficient network utilization in normal operations and robustness to sudden traffic surges as an effect of unforeseen events. We can now define the 2SR-NEP.

Definition 1 (2SR Network Expansion Problem)**.** Given a network graph G, CLs K, traffic demands $\mathbf{T}$, and an MLU threshold U_T (all as previously defined), an instance of the 2SR-NEP is given by $\mathcal{I} = (G, K, \mathbf{T}, U_T)$. The problem asks for an addition vector $y = (y(i))_{i \in I}, y(i) \in \{0, 1\}$, an expansion vector $\lambda = (\lambda(l))_{l \in K}, \lambda(l) \in \mathbb{N}$, and a policy plan $x = (x^k_{s,t})_{s,t,k \in V}, x^k_{s,t} \in \mathbb{Q}_{\geq 0}$, such that (*i*) the cost incurred by implementing y and λ is minimal, and (*ii*) U_T is adhered to when routing $\mathbf{T}$ according to p in the expanded network. $x^k_{s,t}$ describes the fraction of the s-t-demand that should be routed via the midpoint k (following the model of [11]). In an $\mathcal{I}$-solution (y, λ, x), the capacity of any link $l \in B$ is given by $\text{cap}(l) = c(l) + \lambda(l) \cdot \mu(l)$, and the capacity of an innovative link $i \in I$ is given by $\text{cap}(i) = y(i) \cdot \zeta(i) + \lambda(i) \cdot \mu(i)$. We require of every solution that $\text{cap}(l) = \text{cap}(r(l))$ for all $l \in A'$ (cf. Sec. III-A).

We now proceed to prove the problem's NP-hardness. The general idea of the proof is to construct a network in which the existence of a capacity-admissible routing of a demand corresponds to an element being covered in a Set Cover (SC) instance, and expanding the capacity of a link corresponds to choosing a set into the cover. The rest of the construction ensures that certain links cannot be used for certain routing paths, and that a cost-optimal 2SR-NEP solution corresponds to a minimal set cover.

Theorem 1. *The 2SR-NEP is NP-hard.*

Proof. We give a reduction from the NP-hard Set Cover problem. Consider an SC instance $\mathcal{I}_{\text{SC}} = (U, \mathcal{S})$ where $\mathcal{S} = \{S_j \subseteq U : j \in [m]\}$ and $U = \{x_i : i \in [n]\}$. (We use the notation $[m] := \{1, \ldots, m\}, m \in \mathbb{N}$.) We seek a *cover* $\mathcal{C} \subseteq \mathcal{S}$ where $\bigcup_{S \in \mathcal{C}} S = U$ and $|\mathcal{C}|$ is minimal. We construct the 2SR-NEP instance $\mathcal{I}_{\text{NEP}}$ as follows. Define the sets of nodes

$$V_U := \{o_i : i \in [n]\}, V_S := \{\sigma_j : j \in [m]\}, V_d := \{d\},$$

and set $V := V_U \uplus V_S \uplus V_d$. Next, we define the edges of the graph as

$$A_{o \to \sigma} = \{(o_i, \sigma_j) : x_i \in S_j\},\ A_{\sigma \to o} = \{(\sigma_j, o_i) : x_i \in S_j\},$$
$$A_U = A_{o \to \sigma} \uplus A_{\sigma \to o},\ c(l) := 1 \text{ for } l \in A_U, \text{ and}$$
$$A_{\mathcal{S}} := (V_S \times V_d) \uplus (V_d \times V_S),\ c(l) := \frac{1}{|\mathcal{S}| + 1} \text{ for } l \in A_{\mathcal{S}}.$$

Accordingly, $A := A_U \uplus A_{\mathcal{S}}$. Define the CLs as $K = F = A_{\mathcal{S}}$, and let

$$\xi(l) := 0.5 \text{ for } l \in K, \mu(l) := |S_j| \text{ for } (\sigma_j, d), (d, \sigma_j) \in K.$$

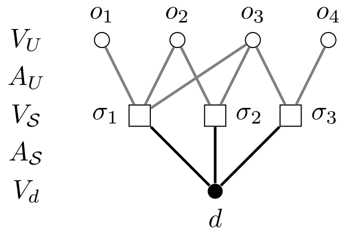

Fig. 1. Topology of 2SR-NEP instance derived from SC instance ($U = \{x_1, x_2, x_3, x_4\}, \mathcal{S} = \{\{x_1, x_2, x_3\}, \{x_2, x_3\}, \{x_3, x_4\}\}$)

We set $w(l) := 1$ for all $l \in A$. The traffic demand from u to v is given by

$$t_{u,v} = t(u, v) := \begin{cases} 1, & (u, v) \in (V_U \times V_d) \cup A_{\sigma \to o}, \\ 0, & \text{otherwise.} \end{cases}$$

Finally, we set the link utilization threshold as $U_T := 1$. An example of this construction is depicted in Fig. 1.

Now, assume that the 2SR-NEP problem can be solved optimally. Let $X \subseteq I$ be the set of links that have been expanded in an optimal solution of $\mathcal{I}_{\text{NEP}}$, i.e.,

$$X := \{l \in K : \lambda(l) > 0\} \subseteq K. \tag{1}$$

Since expanding a forward link implies that the reverse link is expanded as well, $X = \{(\sigma_j, d), (d, \sigma_j) : j \in J^* \subseteq [m]\}$.

Claim. The cover

$$\mathcal{C} := \{S_j : (\sigma_j, d) \in X\} \tag{2}$$

is an optimal solution of $\mathcal{I}_{\text{SC}}$.

To see this, we will first make a few observations.

Observation 1. If $t(\sigma_j, o_i) \neq 0$, the demand is routed via the path $\sigma_j \to o_i$ (i.e., using only the edge (σ_j, o_i)).

To see this, consider the capacity across the cut $\delta(V_S \to V_U)$ and the sum of demand volumes that need to be routed across the cut. Since none of the links in the cut can be expanded by construction, it is easy to see that both quantities are exactly $|A_{\sigma \to o}|$. Therefore, in an admissible solution, the demand is routed via just one link from $\delta(V_S \to V_U)$. Using an additional link would require more capacity than is available in the cut.

Observation 2. In any admissible $\mathcal{I}_{\text{NEP}}$ solution, (at least) one of the links $(\sigma_{j_1}, d), \ldots, (\sigma_{j_k}, d)$ has been expanded.

Consider any non-zero o_i-d-demand and its routing paths $o_i \to \{\sigma_{j_1}, \ldots, \sigma_{j_k}\} \to d$ (the flow may be split). Since $c(l) = 1/(|\mathcal{S}|+1)$ for all $l \in A_{\mathcal{S}}$ per construction, and since there are at most $|\mathcal{S}|$ links to split over, the demand cannot be routed with the existing capacities. The observation follows directly.

Observation 3. In any optimal $\mathcal{I}_{\text{NEP}}$ solution, $\lambda(l) \leq 1$ holds true for all $l \in K$.

After Obs. 1, it suffices to consider the routing of demands $t(\sigma_j, d)$ for $(\sigma_j, d) \in V_{\mathcal{S}} \times V_d$. Per construction, adding one module of capacity to the link (σ_j, d) is enough to satisfy the demand. Since $\xi(\sigma_j, d) > 0$, no excessive capacity is added in an optimal solution.

Observation 4. The cost of X (as defined in (1)) is equal to the cardinality of $\mathcal{C}$ (as defined in (2)).

This is easy to see when considering Obs. 3 and recalling that forward and reverse direction of a link must always be expanded at the same time.

We will now use these observations to show the admissibility and optimality of $\mathcal{C}$ as a solution to $\mathcal{I}_{\text{SC}}$.

On the admissibility of $\mathcal{C}$. We have to show that any $x_i \in U$ is covered by $\mathcal{C}$. In an admissible $\mathcal{I}_{\text{NEP}}$ solution, every non-zero demand has a capacity-admissible routing path. Because of Obs. 1, the links from $A_{\sigma\to o}$ are fully utilized already. Therefore, in every admissible solution, each o_i-d-demand must be split along paths of the form $o_i \to \{\sigma_{j_1}, \dots, \sigma_{j_k}\} \to d$. Because of $(o_i, \sigma_{j_1}), \dots, (o_i, \sigma_{j_k}) \in A$, we know $x_i \in S_{j_1}, \dots, S_{j_k}$. Additionally, according to Obs. 2, one of the edges $(\sigma_{j_1}, d), \dots, (\sigma_{j_k}, d)$ must have been expanded, let us say (σ_{j_1}, d). Because of Obs. 3, $\lambda(\sigma_{j_1}, d) = 1$, implying $(\sigma_{j_1}, d) \in X$, which means $S_{j_1} \in \mathcal{C}$. Hence, $x_i \in S_{j_1} \subseteq \bigcup_{S\in\mathcal{C}} S$, therefore x_i is covered and $\mathcal{C}$ is an admissible solution to $\mathcal{I}_{\text{SC}}$.

On the optimality of $\mathcal{C}$. For the purpose of arriving at a contradiction, assume $\mathcal{C}$ was not an optimal $\mathcal{I}_{\text{SC}}$ solution. Then there exists a subfamily $\mathcal{C}' \subseteq \mathcal{S}$ with $\bigcup_{S\in\mathcal{C}'} S = U$ and $|\mathcal{C}'| < |\mathcal{C}|$. Consider the $\mathcal{I}_{\text{NEP}}$ solution defined as follows:

$$\lambda'(l) := \begin{cases} 1, & l = (\sigma_j, d) \in A_{\mathcal{S}} \text{ and } S_j \in \mathcal{C}', \\ 1, & l = (d, \sigma_j) \in A_{\mathcal{S}} \text{ and } S_j \in \mathcal{C}', \\ 0, & \text{otherwise.} \end{cases}$$

Furthermore, let (*i*) any o_i-d-demand be routed along a path of the form $o_i \to \sigma \to d$, and (*ii*) any σ_j-o_i-demand be routed directly along the edge (σ_j, o_i). It is easy to check that this expansion and policy plan describe an admissible solution to $\mathcal{I}_{\text{NEP}}$ (omitted here for lack of space). Its cost is given by

$$\text{cost}(\lambda') = \sum_{l\in K} (\lambda'(l) \cdot \xi(l)) = \sum_{S_j\in\mathcal{C}'} 2 \cdot (1 \cdot 0.5) = |\mathcal{C}'|.$$

As seen in Obs. 4, $\text{cost}(\lambda) = |\mathcal{C}|$. Therefore,

$$\text{cost}(\lambda') = |\mathcal{C}'| < |\mathcal{C}| = \text{cost}(\lambda).$$

This, however, contradicts the assumption that λ is an optimal $\mathcal{I}_{\text{NEP}}$ solution. Thus, $\mathcal{C}'$ cannot exist and $\mathcal{C}$ is optimal, proving the NP-hardness of $\mathcal{I}_{\text{NEP}}$. □

IV. Algorithmic Approach

With the 2SR-NEP being NP-hard, it is unlikely that there exists an efficient algorithm that provides optimal solutions to this problem for larger instances within reasonable time. However, due to the generally high capital investment required for network expansion, being able to compute solutions that come at least reasonably close to the most cost-effective solution is still an important objective for operators. To address this need, we propose TROPIC, a heuristic algorithm for **TR**affic-Engineering-**O**riented **P**lanning of **I**P **C**ore networks.

One of the biggest hurdles regarding an efficient computation of optimal solutions is the large number of possible innovative links to consider. TROPIC addresses this by employing a two-stage approach. In the first stage, a subset of promising ICLs is heuristically added to the topology. The second stage then computes optimal link expansions for the enlarged network.

$$\min \quad \sum_{i\in I} y(i) \cdot \zeta(i) + \sum_{l\in K\setminus I} \lambda(l) \cdot \xi(l) \tag{3}$$

$$\sum_{u\in V} f^t_{(m,u)} = \sum_{v\in V} f^t_{(v,m)} + t_{m,t} \quad \forall m \neq t \in V \tag{4}$$

$$\sum_{t\in V} f^t_{(u,t)} \leq U_T \cdot \kappa(u,t) \qquad \forall (u,t) \in A \uplus I \tag{5}$$

$$\kappa(l) = \begin{cases} c(l), & l \in A \setminus B, \\ c(l) + \lambda(l) \cdot \mu(l), & l \in B, \\ y(l) \cdot \eta(l), & l \in I \end{cases} \quad \forall l \in A \uplus I \tag{6}$$

$$\kappa(l) = \kappa(r(l)) \qquad \forall e \in K \tag{7}$$

$$f^t_{(u,v)} \in \mathbb{Q}_{\geq 0} \quad \forall (u,v) \in A, t \in V \tag{8}$$

$$y(i) \in [0,1] \qquad \forall i \in I \tag{9}$$

$$\lambda(l) \in \mathbb{Q}_{\geq 0} \qquad \forall l \in K \setminus I \tag{10}$$

Prob. 1. First-stage model of the TROPIC algorithm

A. First Stage: Estimating Relevance of Innovative Candidates

In order to identify promising ICLs, we rely on an MCF-based model. The main reason for using MCF and not SR in this stage is that integrating the concept of link additions into an efficient 2SR formulation would require the use of $\mathcal{O}(2^{|I|})$ configuration variables. Furthermore, it has been shown that 2SR often closely matches MCF solutions in terms of MLU. Hence, even though we assume a perfect routing in the first stage, chances are high that an equivalent routing can be implemented with 2SR as well. To further reduce the computational requirements for this stage, we allow linear (instead of modular) capacity upgrades and allow ICLs to be added "fractionally", i.e., to any degree between 0 and 1. The extents to which ICLs are added in the first-stage solution can be considered an indicator for their respective importance, which is why we use them to select the ICL subset that ought to be added to the network (cf. Sec. IV-B).

Prob. 1 presents the LP formulation for the first stage. The variable $f^t_{(u,v)}$ gives the volume of traffic destined to t that crosses the link (u,v). $y(i)$ represents the degree to which $i \in I$ is built. $\lambda(l)$ defines the amount of capacity that is added to any CL $l \in K \setminus I$. The objective function (3) aims to minimize the total expansion cost, consisting of addition and expansion costs. (4) ensures that every demand is routed in a feasible solution. (5) limits the volume of traffic that can flow over a link such that the MLU threshold is always respected. The available capacity on any link is determined by (6): For non-candidate links, this is simply the capacity in the original network. For non-innovative candidate links, the original capacity can be expanded with additional capacity, controlled via λ. As discussed earlier, ICLs can be built fractionally, but they can never be expanded past their initial capacities η. (7) ensures that capacities in the forward and reverse direction of a link are kept equal (cf. Sec. III-A). Overall, this first stage of TROPIC consists of $\mathcal{O}(|A| \cdot |V|)$ variables and $\mathcal{O}(|V|^2)$ constraints. Therefore, it can be solved in polynomial time with well-known methods (e.g., [25]).

B. Heuristic Link Selection

Using the first-stage solution, $\mathcal{I}$ is modified before being passed to the second stage. The idea of this processing step is to identify valuable ICLs from the addition degrees y. Assuming that links which are cost-effective under an MCF-based routing are also cost-effective under 2SR, all innovative links $i \in I$ with $y(i) \geq \tau$ are added to the network.[2] $\tau \in [0, 1]$ is called the *addition threshold* and is a parameter to be chosen by the network operator. Let $I_\tau^* := \{i \in I : y(i) \geq \tau\}$ be the set of CLs that are selected under τ. The modified problem instance is defined as $\mathcal{I}'_\tau := (G', K', \mathbf{T}, U_T)$, where $G' = (V, A')$, $A' := A \uplus I_\tau^*$, and $K' = B \uplus I_\tau^*$. Notably, $K' \subseteq A'$, i.e., there are no more innovative candidates in the modified candidate set since the selected ICLs have already been added to G'. The costs associated with adding those links is given by $F = \sum_{i \in I_\tau^*} \zeta(i)$, which must be considered when calculating the total costs of the expansion plan. At this point, the shortest paths in the network are recomputed so that the newly added links can be respected in the second stage. $\mathcal{I}'_\tau$ is now used as input to the second-stage model. This link selection step has an asymptotic time complexity of $\mathcal{O}(|I|)$.

C. Second Stage: Exact Capacity Upgrades

After the rounding procedure described in the previous subsection, the candidate set no longer contains innovative links. The goal of the second stage is to optimally expand the links that are present in the topology. For exact capacity upgrades under the 2SR paradigm, we follow the LP formulation proposed in [11]. Since relative link loads for every link can be precomputed when imposing a restriction on the number of allowed segments, such models can be solved more efficiently than more general models. For our second-stage model, we adapt this idea, but change the objective function and extend it with the ability to upgrade link capacities. Prob. 2 shows the LP formulation for the second stage. The routing variable $x_{u,v}^k$ defines the volume of the u-v-demand for which the segment k is installed. In contrast to the first-stage model, the expansion variable $\lambda(l)$ now specifies how many (integer) *modules* of additional capacity are installed on any CL l. The objective function (11) minimizes the total upgrade cost over all CLs. (12) forces the entire u-v-demand to be split over all possible 2SR paths. (13) ensures that the total traffic on any link does not exceed the budget set by the utilization threshold and the installed capacity, which is tracked by (14): Non-candidate links have their capacities fixed at the original volume, and CLs can be expanded with additional modules.

[2]This assumption is supported by the finding that 2SR and MCF often achieve similar TE performance [9], [11].

$$\min \quad F + \sum_{l \in K'} \lambda(l) \cdot \xi(l) \tag{11}$$

$$\sum_{k \in V} x_{u,v}^k \geq t_{u,v} \qquad \forall u, v \in V \tag{12}$$

$$\sum_{u,v \in V} \sum_{k \in V} g_{u,v}^k(l) \cdot x_{u,v}^k \leq U_T \cdot \kappa(l) \qquad \forall l \in A' \tag{13}$$

$$\kappa(l) = \begin{cases} c(l), & l \in A' \setminus K', \\ c(l) + \lambda(l) \cdot \mu(l), & l \in K' \end{cases} \qquad \forall l \in A' \tag{14}$$

$$\kappa(l) = \kappa(r(l)) \qquad \forall l \in A' \tag{15}$$

$$x_{u,v}^k \geq 0 \qquad \forall u, v, k \in V \tag{16}$$

$$\lambda(l) \in \mathbb{N} \qquad \forall l \in K' \tag{17}$$

Prob. 2. Second-stage model of the TROPIC algorithm

Require: $\mathcal{I} = (G, K, \mathbf{T}, U_T)$, addition threshold $\tau \in [0, 1]$

▷ Compute addition degrees for ICLs
$y \leftarrow$ FIRST-STAGE($\mathcal{I}$)

▷ Add promising ICLs to network and update candidate set
$I^* \leftarrow \{i \in I : y(i) \geq \tau\}$, $A' \leftarrow A \uplus I^*$, $K' \leftarrow B \uplus I^*$
$G' \leftarrow (V, A')$, $\mathcal{I}' \leftarrow (G', K', \mathbf{T}, U_T)$

Return result of SECOND-STAGE($\mathcal{I}'$)

Fig. 2. Pseudocode for TROPIC

Links from I that have been added during the post-processing step are now considered as regular, expandable candidates. (15) ensures that all links provide the same capacity in forwards and backwards direction, as in the first stage. This second optimization stage consists of $\mathcal{O}(|V|^3)$ variables and $\mathcal{O}(|V|^2)$ constraints. Although the time complexity is super-polynomial due to the integer variables (17), it can be solved in acceptable time even for large instances, see Sec. VI-C.

Pseudocode for the entire TROPIC algorithm is shown in Fig. 2. When the second-stage optimization finishes, the objective value indicates the total cost of the expansion plan. The final planning result can be extracted from the first- and second-stage optimization variables. The addition plan y can be obtained from the links in I^*, the upgrade plan λ from the values of (17), and the routing policies from (16).

V. Dataset Acquisition

In order to empirically assess the efficacy of the TROPIC algorithm, we require the network topology (Sec. V-A), a traffic forecast for the planning horizon (V-B), and CLs with associated upgrade capacities and costs (V-C). As a service to the research community, we make our evaluation dataset publicly available.[3]

A. Topologies

We base our evaluation dataset on TE problem instances from the publicly available REPETITA dataset [5] which

[3]https://github.com/sys-uos/tropic-instances

Table I
REPETITA INSTANCES CONSIDERED IN THE EVALUATION

Identifier	Name	Nodes	Edges
A	DeutscheTelekom	30	110
B	Digex	31	70
C	CrlNetworkServices	33	76
D	Bics	33	96
E	BtNorthAmerica	33	142
F	Xspedius	34	98
G	Bellcanada	48	128
H	Tinet	48	222
I	HiberniaGlobal	53	160
J	Tw	71	230
K	GtsCe	141	376
L	Cogentco	186	494

features various real-world network topologies (many of them adopted from the Internet Topology Zoo [26]). Out of those, we select a subset of topologies that we deem most representative for ISP backbone networks. In order to be included in our evaluation, a topology must (*i*) be marked with the Internet Topology Zoo tags *Type=COM*, *Backbone=1*, and *Layer=IP*, (*ii*) consist of at least 30 nodes, and (*iii*) have at most 50% nodes with degree 1 as well as a 2-core size [27] of at least 80%. Since the instances of the REPETITA dataset are intended as input to pure TE problems, we modify them to better represent the planning problem we wish to study in this paper. In each topology, parallel links are merged, i.e., capacities on all links between a given node pair are aggregated onto a single link. Further, we collapse nodes that are "unlocated", i.e., those without geographic coordinates. This is necessary as our expansion cost function is based on the geodesic distance between two nodes. When a node is collapsed, it is removed from the topology and its former neighbors are connected in a clique-like manner. The traffic demands originating at, or destined to the collapsed node are redistributed onto the first and penultimate hop of the Shortest Path Routing (SPR) path, respectively. As a final modification, the link capacities are adjusted. In the original instances, all links of a network have the same fixed capacity. In real IP backbones, however, we expect "inner" links to have larger capacities than links on the periphery. To simulate this, we compute the link utilizations under an SPR. Capacities of links that are utilized to more than 70% are expanded such that they exactly fit that threshold. The final selection of instances and their sizes after modification are listed in Table I. The REPETITA dataset additionally contains 5 TMs for each topology, which have been generated according to the Random Gravity Model (RGM) [28]. As all TMs conform to the RGM, we do not expect the choice of the initial TM to have a significant impact on the results. Therefore, we arbitrarily select one for each topology.

B. Traffic Projection

In the RGM, the magnitude of any entry $t_{u,v}$ in the TM is proportional to the product of the magnitudes of the total ingress at node u and the total egress at node v. Under the assumption that the majority of traffic growth can be attributed to an increase in residential customer IP traffic, we believe that population growth models can be a suitable approximation for demand growth in backbone networks. This corresponds nicely to the idea of the RGM which is inspired by trip distribution models developed in the social sciences (e.g., [29]). There, a log-normal distribution is often used to model city population growth [30], [31]. We also follow this approach and parameterize a log-normal distribution to model xgress volume growth rates for each Point of Presence. The parameters μ and σ^2 of the underlying normal distribution are chosen according to a maximum-likelihood estimation based on historical year-on-year busy-hour traffic growth rates [1], [9]. This distribution is used to sample xgress growth rates for each node, which are then translated back into traffic demands according to the gravity model.

C. Candidate Links

In most cases, it is trivial to add more capacity to an existing link by either injecting additional wavelengths into the fiber, or by replacing a previous-generation optical module with a more recent one. Therefore, we allow all existing network links to be upgraded in our evaluation dataset; i.e., using the notation introduced earlier, $B \coloneqq A$. Regarding innovative candidates, we only consider links with a maximum length of 500 km, motivated by the approximate distance modern fiber-optical hardware can span using erbium-doped fiber amplifiers [32].

1) Selection Heuristic: Since there is no *a priori* knowledge about which candidate set size performs well, and to limit the computational cost, we decide to bound the number of allowed innovative candidates by the number of existing links, i.e., $|I| \leq |A|$. To make a selection from the set of potential candidates, we developed the Diverse Shortest Paths Score (DSPS) heuristic. Let $\sigma_G(u)$ be the set of shortest paths that run through node u in G. For two nodes $u, v \in V$, their DSPS is defined as follows:

$$\text{DSPS}(u,v) \coloneqq |\sigma_G(u) \setminus \sigma_G(v)| \cdot |\sigma_G(v) \setminus \sigma_G(u)|$$

Our selection heuristic computes the DSPS for all node pairs $(u, v) \in V^2 \setminus A$ that are less than 500 km apart and ranks them in a non-ascending manner. From this list, the first up to $|A|$ pairs are selected as ICLs. Intuitively, the DSPS heuristic attempts to incorporate information from the existing IGP to identify node pairs that connect different parts of the network. Overall, in our dataset, the candidate set for any network $G = (V, A)$ can be written as $K = A \uplus I$ where $|I| \leq |A|$.

2) Cost, Capacity, and Routing Weight: For each link in K, we must also define an expansion capacity and the associated expansion cost. For innovative links, an initial capacity and addition cost must also be defined. We determine the capacity provided by a single module by projecting the technological advancement since the last planning cycle 5 years ago. We assume that the median capacity of links in the base topology corresponds to one module. The compound annual growth rate of fiber-optical capacity sits relatively steady at 15% [33], [34]. The planning capacity we assume is therefore $1.15^5 \approx 2$ times

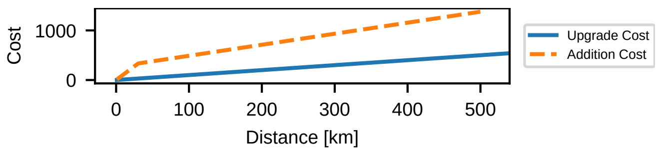

Fig. 3. Cost functions used in evaluation dataset (after normalization): $\text{cost}_{\text{upgrade}}(l) \mapsto l$ and $\text{cost}_{\text{add}}(l) \mapsto \langle(11.11 \cdot l)$ for $l \leq 30; (11.11 \cdot 30 + (l - 30) \cdot 2.22)$ for $l > 30\rangle$. (Assume the length of a CL is given as l km.)

the median link capacity in the base topology. In order to determine the associated expansion costs, we use two distance-based cost functions: one for addition, and one for upgrade costs. Fig. 3 plots both functions. They reflect the TCO of adding one module of capacity at a given length. The addition cost function consists of two linear segments to account for the high bring-up costs of installing a new link at all.

The routing weights of added links are set to the median link weight in the base topology. Since we use the unary-weighted topologies of the REPETITA dataset (cf. Sec. V-A), innovative candidates are always added with an IGP weight of 1.

VI. Evaluation

Sec. VI-A presents the algorithmic parameters as well as the hardware environment. In Sec. VI-B, we describe a baseline algorithm to compare our results against. The evaluation results are reported and discussed in Sec. VI-C and VI-D.

A. Hardware and Software Parameters

In our evaluation, we set $U_T = 0.7$ for all instances. TROPIC additionally requires specification of the addition threshold τ (cf. Sec. IV-B). For our evaluation, we set $\tau = 0.5$ with the intuition that at least half of an innovative link's capacity must be added in the first stage to justify adding it to the network. We implemented TROPIC in C++ using CPLEX [35]. The evaluation was performed on a machine with an AMD EPYC 7452 processor and 250 GiB of memory. Both algorithms are evaluated on all instances listed in Table I, each with two candidate sets: one with *only existing* (OE) links, and one additionally including ICLs as determined by the DSPS heuristic (cf. Sec. V-C). For instance L, ICLs were also chosen according to DSPS, however, due to memory constraints the number of innovative links had to be reduced from 496 to 248.

B. Baseline Solution

In order to evaluate the performance of TROPIC, we require a baseline solution to compare against. For this purpose, we designed a greedy capacity expansion algorithm that is in line with what a network planner would intuitively do by hand [36]. We call this algorithm GreedySPR-Planning (GSPR, see pseudocode in Fig. 4). The idea is to iteratively introduce additional capacity to the network in the most opportune places. In every iteration, the next link to add capacity on is determined by its *value*, relating the amount of removed overutilization to the cost of removing it. The routing paths in the network are never modified by GSPR, except when the addition of an ICL changes the shortest-path set between node pairs. The SPR includes equal-cost multi-path splitting.

Require: $\mathcal{I} = (G, K, \mathbf{T}, U_T)$
while a link l with $\text{util}(l) > U_T$ exists **do**
  $m \leftarrow$ any MLU link in K
  ▷ $A \oplus o$: link set after candidate option o is applied
  ▷ Compute values of available candidate options
  for all $o \in \{m\} \cup \{i \in I : i$ not added already$\}$ **do**
    $V_o \leftarrow \frac{1}{\text{cost}(o)} \cdot \Big(\sum_{l \in A, \text{util}_A(l) > U_T} (\text{util}_A(l) - U_T)$
      $- \sum_{l \in A \oplus o, \text{util}_{A \oplus o}(l) > U_T} (\text{util}_{A \oplus o}(l) - U_T)\Big)$
  if $V_m \geq V_i$ for all i **then**
    $A \leftarrow A \oplus m$ ▷ Minimally upgrade m and $r(m)$
  else
    $i^* \leftarrow \arg\max_i V_i$
    $A \leftarrow A \oplus i^*$ ▷ Add and min. upgrade i^* and $r(i^*)$
    Recompute SPR and link loads

Fig. 4. Pseudocode for GSPR

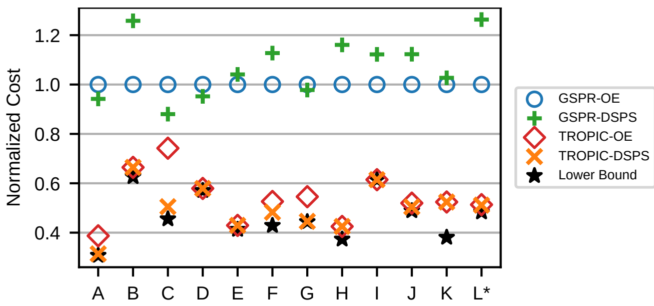

Fig. 5. Normalized solution costs for instances in the REPETITA dataset

C. Results

Fig. 5 shows solution costs for the REPETITA instances. To allow for easier cross-instance comparison, the costs have been normalized to those of the GSPR-OE solution in each topology. In addition to the baseline solutions, we have also computed a lower bound for each instance. This bound is derived by solving an exact version of TROPIC's first stage for the DSPS candidate set: link additions and expansions are integral, and added innovative links may be expanded after they have been added. The demands are still routed according to an MCF, which makes these solutions infeasible to implement in real-world networks. Due to its computational complexity, this lower-bound LP could not be solved to optimality for instances J and L. For these instances, the reported lower bounds are the best ones found by CPLEX before running out of memory. Similarly, for instance L, TROPIC's costs and solve times refer to solutions with an optimality gap of less than 2%.

All but the three largest instances can be solved by TROPIC in under 1 hour. Instance J requires a solve time of 4 days with the DSPS candidate set, whereas K and L took roughly 30 days. GSPR is much faster, requiring at most 1 hour even on the largest instances. The cost reductions offered by TROPIC range between 25% and 68%, with the majority located in the region between 40% and 60%. Allowing ICLs improves TROPIC's solution quality in 5 of 12 instances. Importantly, in all instances, the costs *never increase* through the inclusion of innovative candidates. This is in sharp contrast to GSPR, where enlarging the candidate set can significantly degrade the

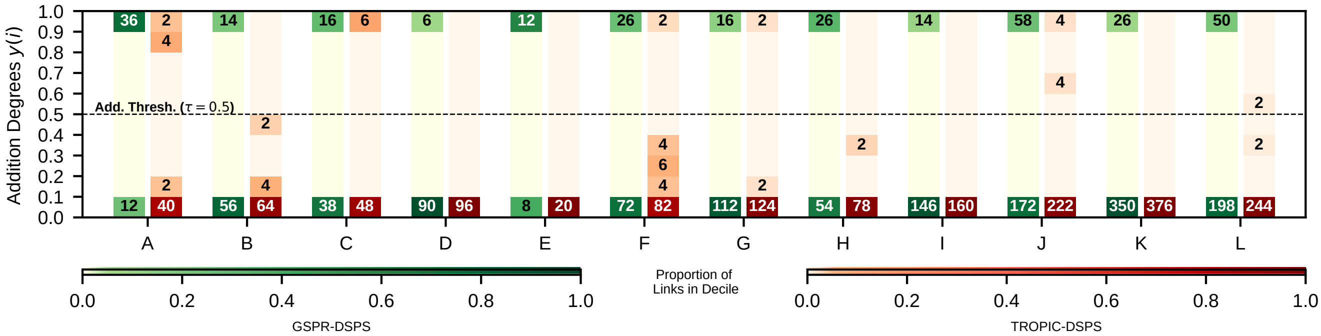

Fig. 6. Distribution heatmap of addition degrees for the DSPS instances of the REPETITA dataset. The addition degrees of ICLs are binned into deciles. The color of a decile box indicates the proportion of links inside that decile. Each box is annotated with the absolute number of links contained; an empty box indicates that no links fall into that decile. The vast majority of observations in the first deciles $[0, 0.1)$ are exactly 0.

solution quality, e.g., in instances B and L. This is an attractive property of TROPIC: An operator can provide a large set of CLs without concern for the deterioration of solution quality.

Fig. 6 depicts the addition degrees of innovative links in each DSPS solution. For the TROPIC solutions, this refers to the $y(i)$ values after the first stage. For the GSPR solutions, we interpreted innovative links to have a degree of 1 if they were added in some iteration of the algorithm, and 0 if not. The total number of added links can be obtained by summing all numbers above the threshold line. We see that TROPIC is much more conservative at adding innovative links than GSPR, especially in the larger instances J and L, but also in the smallest instance A. We argue that there are two possible reasons why TROPIC uses innovative links so infrequently: (*i*) The candidate strategy yields a suboptimal selection of innovative links, or (*ii*) the initial topology already includes all worthwhile links and there are no valuable links left to be added. In contrast, GSPR adds links quite optimistically. This reveals a key weakness of the greedy approach: It is unable to distinguish locally-good from globally-good innovative links. The MCF-based first stage of TROPIC, on the other hand, does not appear to suffer from the same problem: If innovative links *are* used in a TROPIC-DSPS solution, the resulting solution costs are always less than the corresponding TROPIC-OE solution (cf. Fig. 5). We consider this an indication that the MCF-based first-stage heuristic works well in the context of selecting promising links for 2SR. This is further backed up by the deviations of the TROPIC-DSPS costs from the computed lower bounds. The largest deviation from the bound is found in instance K, where the lower bound is 0.143 below the normalized cost of the solution found by TROPIC. In nearly all other instances, the solutions produced by TROPIC come exceedingly close to or even achieve the lower bound. This impressively illustrates the effectiveness of 2SR for TE as well as the heuristic quality of the MCF-based first stage.

Finally, it is worth noting that the number of links added by TROPIC does not only depend on the values $y(i)$ in the first stage, but also on τ. Fig. 6 gives a rough indication of how the number of added links would change with different values for τ. For most instances, no large changes would occur as long as $\tau > 0$. This implies that TROPIC is relatively robust towards τ, except for extremely large or small values.

D. Discussion

1) Baseline Solutions: Ideally, TROPIC should be compared to other sophisticated capacity planning approaches (cf. Sec. II). Unfortunately, there are no publicly available implementations of these approaches, and reimplementing them from scratch is not feasible in the context of this paper. Therefore, we use GSPR as a baseline approach as it mimics a network evolution process still considered best practice [36]. It can be argued that GSPR is an unfair point of comparison as only SPR is used instead of the more powerful 2SR available to TROPIC. To address this, we also investigated G2SR; a greedy algorithm which computes an MLU-optimal 2SR plan for every expansion option before making the greedy choice. Although G2SR can achieve moderate cost improvements over GSPR, its solutions are still substantially more expensive than those produced by TROPIC. Crucially, we observed that G2SR does not scale well with instance size. For example, even on the smallest DSPS instance of our dataset (instance A), G2SR requires 53 min (!) to produce a solution with a normalized cost of 0.77. In comparison, GSPR takes 1.7 sec for a cost of 0.94, and TROPIC takes 9.6 sec for a cost of 0.31. Based on these and similar initial results, we decided not to investigate G2SR any further and focused on TROPIC.

In this context, we stress the importance of the lower-bound solutions discussed in Sec. VI-C. Although we do not have reference costs of other state-of-the-art approaches, the median cost degradation of TROPIC to the lower bound is just 4.9%, and in many cases, nearly 0% (thus quasi-optimal, cf. Fig. 5).

2) Candidate Link Sets: All reported results refer to a specific candidate set (in our case, either DSPS or OE). It can not be ruled out that there may be candidate sets on which a given algorithm performs better or worse than on the studied sets. However, since there are on the order of $\mathcal{O}(2^{|V|^2})$ possible candidate sets, it is infeasible to compute solutions for all of them on larger instances. In the evaluation, the DSPS heuristic was used to find ICLs, but other strategies may be conceived. A broader investigation may be part of future work.

Additionally, in a real-world scenario, many links might not be considered as candidates due to external factors, e.g., topographical constraints, vendor compatibility, or legal restrictions. The model proposed in Sec. III-B already allows arbitrary links to be excluded from K, providing network operators with full flexibility to capture the specific limitations of their networks. While it is not possible to add such constraints to the REPETITA instances due to limited public data, it is important to stress that operators would be able to use knowledge about their own networks in order to determine viable candidates in a real-world use case.

3) Solve Times: While TROPIC mostly computes its solutions in less than 1 hour, up to 30 days of computation time can be required for the largest instances K and L (cf. Sec. VI-C). Even though this number is quite large, it does not impose a meaningful limitation regarding the suitability of TROPIC for its intended use case of long-term capacity planning. In this context, we are generally considering planning horizons of multiple years. Furthermore, solve times also depend on the respective candidate set K. In particular, shrinking K can considerably reduce computation times (at the potential cost of reduced solution quality).

VII. Conclusion

Cost-efficient network expansion is one of the most delicate problems concerning today's network operators. It has proven beneficial to explicitly consider the routing of demands in the future expanded network, but existing approaches do not make use of state-of-the-art SR-TE techniques or are infeasible to implement in practice. In this paper, we introduced the problem of cost-effectively expanding IP networks while employing a 2SR scheme that should be implementable in most modern backbones. Since solving the problem is NP-hard, we propose TROPIC; a heuristic algorithm that simultaneously considers SR-TE and network expansion through capacity upgrades and link additions. In an extensive evaluation using representative network topologies derived from the publicly available REPETITA dataset, we show that, despite its heuristic nature, TROPIC is able to considerably outperform baseline algorithms, enabling cost savings in the range of 40–60%. For most instances, it even comes close to a theoretical lower bound. These findings not only demonstrate the capabilities of our TROPIC approach, but more generally underline the importance of considering the objectives of network expansion and TE jointly, instead of treating them as independent problems, as is still common practice to date. Hence, our future work will involve a detailed analysis of further real-world constraints, e.g., the robustness to uncertain traffic projections, limited demand splitting, or reducing the number of required SR policies.

References

[1] Cisco, "Visual Networking Index, 2017–2022," Tech. Rep., 2019. [Online]. Available: https://twiki.cern.ch/twiki/pub/HEPIX/TechwatchNetwork/HtwNetworkDocuments/white-paper-c11-741490.pdf

[2] Nokia, "Global network traffic report," Tech. Rep., 2023. [Online]. Available: https://onestore.nokia.com/asset/213660

[3] Cisco, "WAE design 7.6.x user guide," Tech. Rep., 2022. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/net_mgmt/wae/7-6-0/design_user_guide/cisco-wae-design-76-user-guide.html

[4] Juniper Networks, "NorthStar planner user guide," Tech. Rep., 2023. [Online]. Available: https://www.juniper.net/documentation/us/en/software/northstar6.2.6/northstar-planner-user-guide/index.html

[5] S. Gay *et al.*, "REPETITA: Repeatable experiments for performance evaluation of traffic-engineering algorithms," 2017. [Online]. Available: arXiv:1710.08665

[6] C. Filsfils *et al.*, "The segment routing architecture," in *Proc. GLOBECOM*, 2015.

[7] Cisco, "What is segment routing?" https://www.cisco.com/c/en/us/solutions/segment-routing.html.

[8] Juniper Networks, "What is segment routing?" https://www.juniper.net/us/en/research-topics/what-is-segment-routing.html.

[9] T. Schüller *et al.*, "Traffic engineering using segment routing and considering requirements of a carrier IP network," *IEEE/ACM ToN*, Aug. 2018.

[10] P. L. Ventre *et al.*, "Segment routing: A comprehensive survey of research activities, standardization efforts, and implementation results," *IEEE Commun. Surv. & Tut.*, 1st Quarter 2021.

[11] R. Bhatia *et al.*, "Optimized network traffic engineering using segment routing," in *Proc. INFOCOM*, 2015.

[12] M. Jadin *et al.*, "CG4SR: Near optimal traffic engineering for segment routing with column generation," in *Proc. INFOCOM*, 2019.

[13] R. Hartert *et al.*, "A declarative and expressive approach to control forwarding paths in carrier-grade networks," *SIGCOMM CCR*, Aug. 2015.

[14] S. Gay *et al.*, "Expect the unexpected: Sub-second optimization for segment routing," in *Proc. INFOCOM*, 2017.

[15] F. Hao *et al.*, "Optimizing restoration with segment routing," in *Proc. INFOCOM*, 2016.

[16] T. Schüller *et al.*, "Failure resiliency with only a few tunnels – enabling segment routing for traffic engineering," *IEEE/ACM ToN*, Feb. 2021.

[17] Y. Tian *et al.*, "Traffic engineering with segment routing considering probabilistic failures," in *Proc. CNSM*, 2021.

[18] M. Pióro and D. Medhi, *Routing, flow, and capacity design in communication and computer networks*. Elsevier Science, 2004.

[19] S. S. Ahuja *et al.*, "Capacity-efficient and uncertainty-resilient backbone network planning with Hose," in *Proc. SIGCOMM*, 2021.

[20] K. Holmberg and D. Yuan, "Optimization of Internet Protocol network design and routing," *Networks*, Jan. 2004.

[21] O. Gerstel *et al.*, "Multi-layer capacity planning for IP-optical networks," *IEEE Commun. Mag.*, Jan. 2014.

[22] V. Gkamas *et al.*, "A joint multi-layer planning algorithm for IP over flexible optical networks," *J. Lightw. Technol.*, Jul. 2015.

[23] S. Zhang and U. Killat, "Multiple-layer network planning with scenario-based traffic forecast," in *Proc. 17th EUNICE Workshop Energy-Aware Commun.*, 2011.

[24] B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS weights in a changing world," *IEEE JSAC*, May 2002.

[25] N. Karmarkar, "A new polynomial-time algorithm for linear programming," in *Proc. STOC*, 1984.

[26] S. Knight *et al.*, "The Internet Topology Zoo," *IEEE JSAC*, Oct. 2011.

[27] S. B. Seidman, "Network structure and minimum degree," *Social Netw.*, Sep. 1983.

[28] M. Roughan, "Simplifying the synthesis of Internet traffic matrices," *SIGCOMM CCR*, Oct. 2005.

[29] D. M. Levinson and A. Kumar, "A multi-modal trip distribution model," *Transp. Res. Rec.*, 1994.

[30] J. Eeckhout, "Gibrat's law for (all) cities," *Amer. Econ. Rev.*, Dec. 2004.

[31] R. González-Val *et al.*, "New evidence on Gibrat's law for cities," *Urban Stud.*, Jan. 2014.

[32] OpenZR+ MSA, "OpenZR+: 400G digital coherent optics for multi-haul," Tech. Rep., 2020. [Online]. Available: https://openzrplus.org/site/assets/files/1074/openzrplus_whitepaper_-_sept_29_2020_final.pdf

[33] P. J. Winzer *et al.*, "Fiber-optic transmission and networking: the previous 20 and the next 20 years," *Opt. Express*, Sep. 2018.

[34] X. Liu, "Evolution of fiber-optic transmission and networking toward the 5G era," *iScience*, Dec. 2019.

[35] IBM, "ILOG CPLEX Optimization Studio 20.1.0," https://ibm.com/docs/en/icos/20.1.0, 2020.

[36] Cisco, "Best practices in core network capacity planning," Tech. Rep., 2020. [Online]. Available: https://www.cisco.com/c/en/us/products/collateral/routers/wan-automation-engine/white_paper_c11-728551.pdf