The Case for Stochastic Online Segment Routing under Demand Uncertainty

Jérôme De Boeck
University of Fribourg

Bernard Fortz
Université libre de Bruxelles & INRIA

Stefan Schmid
TU Berlin & Fraunhofer SIT

***Abstract*—Segment routing has recently received much attention in industry and academia for providing simple yet powerful and scalable traffic engineering, a most important concern for Internet Service Providers. However, the fundamental optimization problem underlying segment routing needs to be better understood today. This paper addresses this gap and presents a novel algorithmic approach to optimize traffic engineering in segment routing networks, accounting for demand uncertainty. In particular, we propose a stochastic approach to online segment routing which uses a conditional value at risk when accounting for the traffic matrix uncertainty. This approach can perform significantly better than the worst-case approach often considered in the literature. We also show that depending on the demand volatility, our stochastic approach can be further optimized in that it is sufficient to account for only a part of the demand without sacrificing traffic engineering quality.**

***Index Terms*—Traffic engineering, segment routing, optimization, uncertainty**

I. Introduction

To make efficient use of their infrastructure and to avoid congestion, most major Internet Service Providers (ISPs) today employ sophisticated algorithms for *traffic engineering (TE)*, i.e., to steer intra-domain traffic through their network optimally. Indeed, given the explosive growth of communication traffic, especially to and from data centers, due to the popularity of data-centric applications, TE is a crucial concern of ISPs, and many innovations in networking over the last years were at least partially motivated by the desire to improve TE [1], [2]. Traditionally, TE was relatively inflexible as routes could only be influenced *indirectly* via link weight optimization. Moreover, the network operator has no direct control over the repartition of flows on multiple shortest paths, as this repartition is usually handled using Equal Cost Multi-Paths rules (ECMP).

Segment routing (SR) is an emerging networking architecture developed within the IETF, which introduces a novel variant of source routing in which traffic can be steered away from congested shortest paths by inserting intermediate destinations, so-called waypoints. More specifically, in a segment-routed network, a source node s can prepend one or multiple waypoints w to packet headers, which will be traversed between the source s and destination d. In each such segment, packets are forwarded along the shortest path. Segment routing hence provides an additional dimension for optimization: waypoints can also be optimized in addition to the link weights. Empirical studies show that such a redirection, already with a single waypoint, can significantly improve TE [3], [4]. SR is also attractive in that, compared to other approaches such as MPLS, it only requires maintaining the path state on the ingress node and allows for a significantly simpler control plane [5].

This paper considers the segment routing problem from a fundamental algorithmic perspective. In particular, we are interested in the question of how segment routing can be optimized under demand uncertainty, a significant practical concern: in many practical deployments, traffic demands change over time and are not always perfectly predictable, e.g., due to optimizations by content providers [6], [7]. The segment routing mechanism hence cannot be tailored toward a given scenario but should also account for variability. In particular, it should balance the benefits of optimizing the routes toward the demand and the stability benefits of not rerouting the traffic frequently.

This paper makes a case for a *stochastic* approach to online segment routing. Given a possible demand polytope or set of traffic matrices, we propose considering the conditional value at risk to account for the uncertainty. While this approach is still very conservative, it goes beyond the common methods revolving around the worst case.

While our model and approach are more general, for our empirical evaluation, we focus on a single waypoint throughout this paper and to compare it to state-of-the-art (so: two segments). It has been shown that most benefits of segment routing can be reaped already in such a scenario [5].

Our stochastic approach can perform significantly better than the worst-case approach often considered in the literature. We also show that depending on the demand volatility, our stochastic approach allows for further interesting optimizations in that it is sufficient to only account for a part of the demand space without sacrificing TE quality.

The remainder of this paper is organized as follows. Section II provides an overview of related work on TE using segment routing. In Section III, we introduce the problem definition and generic formulations of segment routing, which are used in the online procedure described in Section IV. The robust segment routing algorithm using a single waypoint of Bhatia et al. [5] and the formulations we use in our online procedure are presented in Section V. In Section VI, we present numerical experiments illustrating the benefits of a stochastic approach over a robust approach. Conclusions are presented in Section VII.

ISBN 978-3-903176-57-7 ©2023 IFIP

II. Related Work

There already exists a large body of literature on TE, for various network protocols, from MPLS [8] over SDN [9] to segment routing [10]. There are several well-known analytical and complexity results on the achievable performance of TE, e.g., by [11], [12] and others [12], [13],

Many results on TE for segment routing are already available in e.g. [3], [5], [14], [15], see [4] for a recent survey. [16] considers the problem of determining the optimal parameters for segment routing in offline and online scenarios; the authors present a game-theoretic analysis and propose an oblivious segment routing approach. Moreno et al. [15] showed that, interestingly, a minimal number of stacked labels suffice to exploit the benefits of segment routing successfully. Aubry et al. [3], [17] lay the algorithmic foundations of segment routing, considering different and related applications. Recently, Parham et al. [1] present an analytical quantification of the benefits of joint waypoint and weight optimization with segment routing. Bhatia et al. [5] develop a traffic matrix oblivious algorithm for robust segment routing in the offline case. However, we are unaware of any work on stochastic approaches to deal with uncertainty in segment routing TE.

III. Problem description and generic models

We consider a directed network described by a graph $G = (V, A)$ where V is the set of nodes and A is the set of arcs, and each arc $a \in A$ has a given capacity $c_a > 0$. A static routing strategy is also given (in our experiments, we assume OSPF with ECMP and inverse capacity metrics, but our results hold for any routing pattern), as well as a set of traffic matrices $\mathcal{D}$ that represent different potential scenarios for the evolution of the traffic. Each traffic matrix $D \in \mathcal{D}$, also called Traffic Matrix (TM), has an entry d_{ij}^D for each pair $(i, j) \in V^2$ representing the amount of flow requested between i and j. These matrices might be based, for example, on historical data. Our objective is to use segment routing to improve TE for this set of matrices.

We represent a routing by a vector $x \in X^w$ where X^w is the set of all feasible routings using w waypoints (corresponding to $w + 1$ segments, i.e., the w waypoints plus the final destination). A routing is feasible if all the demand is routed through the network for each pair $(i, j) \in V^2$. Vector x defines the quantity routed for each pair $(i, j) \in V^2$ through w waypoints and lies in $[0, 1]^{n^{w+2}}$. Several TE measures $\mu^D(x)$ can be used to evaluate the quality of a routing x and a given TM $D \in \mathcal{D}$ [18]. A common TE measure is the maximum link utilization providing an upper bound on the congestion on a network. A generic formulation for segment routing using w waypoints on a given TM D can be written as follow:

(SR-w)

$$\min \quad \mu^D(x) \tag{1a}$$

$$\text{s.t.} \quad x \in X^w \tag{1b}$$

A. Robust optimization

The first approach is to develop a fully robust solution to the problem, finding a good set of waypoints for all the TMs in $\mathcal{D}$. We introduce variable Θ to represent the resulting worst-case TE measure. A linear program that achieves this purpose (R-SR-w) is presented below.

(R-SR-w)

$$\min \quad \Theta \tag{2a}$$

$$\text{s.t.} \quad \mu^D(x) \le \Theta \quad D \in \mathcal{D} \tag{2b}$$

$$x \in X^w \tag{2c}$$

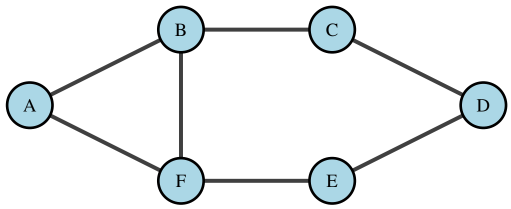
Inequalities (2b) force Θ to be equal to the worst TE measure in an optimal solution.

To illustrate the benefits of such an approach, consider the example network in Figure 1a, where we assume all links have a capacity equal to 10. Consider 3 demands in this network, from A to D, A to E, and B to E. Using OSPF with ECMP on uniform weights would result in the routing shown in Figure 1b, where the demand from A to D is split evenly on paths A-B-C-D and A-F-E-D. Suppose the three demands have a volume of 4. In that case, the resulting flows lead to a maximum link utilization of 1 reached on link F-E (the flow on F-E is composed of 4 units of demand A-E, 4 units of demand B-E, and 2 units of demand A-D as this last one is split due to ECMP, so the capacity of 10 is exactly reached). Using a single waypoint for demand A-E, forcing it to go through C, leads to the routing shown in Figure 1c where the maximum utilization is 0.6, reached on links B-C and F-E, a much better solution. Finally, consider a shift in demand where the volume of demand A-D is still 4, but demand A-E is now of 2 units while demand B-E becomes 8 units. For this new TM, both OSPF and the addition of segment A-C-E have a maximum utilization of 1. Adding the segment path A-B-D, as illustrated in Figure 1d, reduces the maximum utilization to 0.8. Moreover, this solution is more robust than the previous one as the maximum link utilization for the initial TM is also 0.8, hence better than the original OSPF solution.

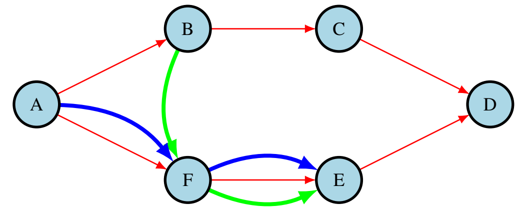
However, this robust model is very conservative. Some matrices in $\mathcal{D}$ might represent extremely bad situations with a very low appearance rate. In this case, the average case may deteriorate too much with a robust approach. On the other hand, we still want some protection against bad cases. The following subsection presents a model based on the concept of Conditional Value-at-Risk as a better compromise between nominal behavior and worst-case situations.

B. Stochastic optimization

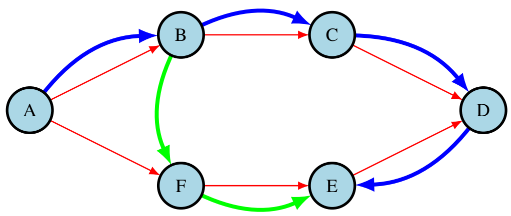
There is a vast literature on the development of optimization models aimed at controlling the trade-off between risk and reward, especially in finance. Conditional value-at-risk (CVaR), first introduced in [19], has probably become the most popular risk measure over the last decades. Optimization models integrating CVaR to control risk have been applied in



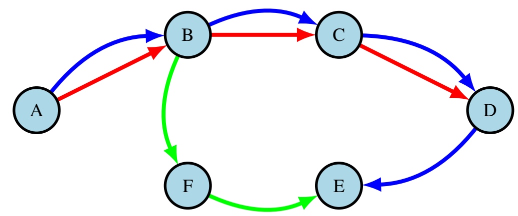
(a) Example network



(b) OSPF routing



(c) OSPF routing + segment A-C-E



(d) OSPF routing + segments A-C-E and A-B-D

Fig. 1: An illustration of several routing strategies

several application domains beyond financial optimization, see e.g. [20] for a recent survey.

In our context, let us assume that each TM $d \in \mathcal{D}$ appears with a given probability p_D. For a given assignment of waypoints to demands, let us define U as the random variable representing the TE measure over matrices randomly picked in $\mathcal{D}$, and let $F_U(z) = \mathcal{P}(U \leq z)$ be its cumulative distribution function. For a given confidence level $\epsilon \in [0, 1)$, the Value at Risk of U is the ϵ-quantile, i.e.

$$\text{VaR}_\epsilon(U) = \min\{z | F_U(z) \geq \epsilon\}.$$

The CVaR of U measures the conditional expectation of U given that $U \geq VaR_\epsilon(U)$, i.e.,

$$\text{CVaR}_\epsilon(U) = \mathbb{E}[U | U \geq \text{Var}_\epsilon(U)].$$

Following the fundamental result of [19], we can compute the CVaR as the solution to a minimization problem, that is,

$$\text{CVaR}_\epsilon(U) = \min_z \left\{ z + \frac{1}{1-\epsilon} \sum_{d \in \mathcal{D}} p_D (\Theta^D - z)^+ \right\}$$

The parameter ϵ controls the aversion to risk. The extreme case $\epsilon = 0$ corresponds to no aversion to risk at all, i.e., the objective is equivalent to the expected value of the TE measure. As ϵ tends to 1, the objective tends to a fully conservative solution, i.e., the solution presented in the previous subsection.

Integrating this objective in a linear program is now straightforward and leads to the stochastic model (S-SR-w):

(S-SR-w)

$$\min \quad z + \frac{1}{1-\epsilon} \sum_{D \in \mathcal{D}} p^D y^D \tag{3a}$$

$$\text{s.t.} \quad y^D \geq \mu^D(x) - z \qquad D \in \mathcal{D} \tag{3b}$$

$$y^D \geq 0 \qquad D \in \mathcal{D} \tag{3c}$$

$$x \in X^w \tag{3d}$$

The auxiliary variables y^D are used to model the non-linear term $(\mu^D(x) - z)^+$ and their values are set by constraints (3b) and (3c).

IV. Online Routing

The generic formulations proposed in the previous section need a set of TMs $\mathcal{D}$ over which segment routing is optimized. The segment routing defined at a given point in time might not be efficient for TMs not considered in $\mathcal{D}$, which might occur after the routing is already defined. To address this situation, we propose a robust and a stochastic online procedure optimizing the routing each time a new TM is available, typically a couple of times per day for internet service providers [21].

In the robust situation, the routing is optimized only when a new worse-case TM is observed. A traffic matrix D is a new worse case if the TE measure $\mu^D(x)$ computed based on the current segment routing $x \in X^w$ is greater than the last optimal value of R-SR-w. If this happens, the TM is added to $\mathcal{D}$. This prevents the model from considering all TMs observed. Figure 2 provides a flow chart of the Robust Online segment routing (RO).

In the stochastic situation, each new TM is added to $\mathcal{D}$ before reoptimizing S-SR-w. Figure 3 provides a flow chart of the Stochastic Online segment routing (SO).

Unlike RO, SO adds each TM to $\mathcal{D}$, potentially leading to a larger formulation for the routing problem. In order to limit

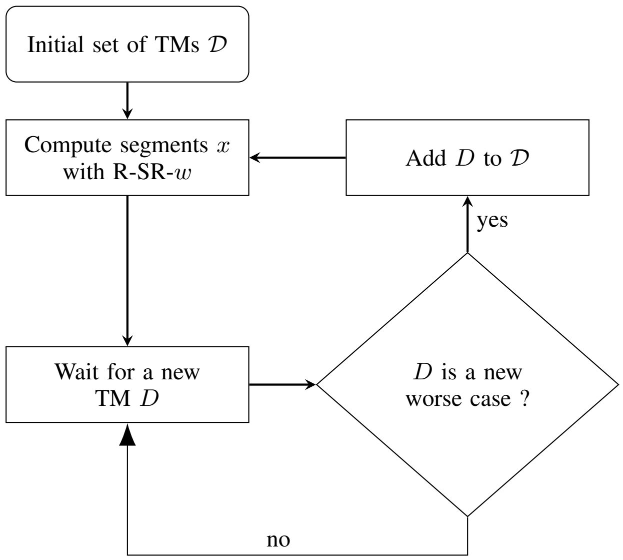


Fig. 2: Robust online SR flowchart

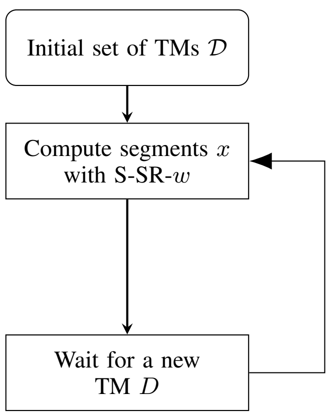


Fig. 3: Stochastic online SR flowchart

the size of S-SR-w, we consider restricting the TMs used in procedure SO-n. As TMs have a certain stability through time, the last n TMs observed are kept in with a probability $\frac{1}{2n}$, and the former n matrices are averaged in a central TM with an associated probability of $\frac{1}{2}$. The number of TMs in $\mathcal{D}$ in procedure SO-n is thus equal to $n+1$, the n last TM observed, and an additional central TM.

V. Segment routing with one waypoint

In this paper, we consider the frequently used Maximum Link Utilization (MLU) as TE measure and a single waypoint to use in the formulations presented in the previous sections for comparison purposes with the oblivious model of Bhatia et al. [5]. Still, any additive convex function in the loads on the arcs or any number of waypoints might be used without changing our methodology. The formulation of Bhatia et al. [5] considering a single waypoint and minimizing the MLU Θ for a predetermined TM D is the following:

(SR-1)

$$\min \quad \Theta \tag{4a}$$

$$\text{s.t.} \quad \sum_{k \in V} x_{ij}^k = 1 \qquad (i,j) \in V^2 \tag{4b}$$

$$\sum_{(i,j) \in V^2} d_{ij}^D \sum_{k \in V} g_{ij}^k(a) x_{ij}^k \leq c_a \Theta \qquad a \in A, \tag{4c}$$

$$x_{ij}^k \geq 0 \qquad (i,j,k) \in V^3 \tag{4d}$$

The demand from i to j in D is denoted d_{ij}^D and x_{ij}^k is the proportion of this demand routed through waypoint k. Note that by convention, we can use $k = j$ if no waypoint is used. Given a fixed underlying routing strategy such as OSPF, we denote $g_{ij}^k(a)$, the fraction of the flow from i to j that would be sent on arc a if waypoint k is used, for all $i, j, k \in V$ and $a \in A$. These values can be easily pre-computed [5]. Constraint (4b) ensures all the demand from each pair $(i,j) \in V^2$ is routed through the network. Combined with the minimizing objective function, constraint (4c) ensures the MLU is properly Θ computed.

From formulation SR-1, Bhatia et al. derive a formulation for an oblivious TM:

(O-SR-1)

$$\min \quad \Theta \tag{5a}$$

$$\text{s.t.} \quad \sum_{k \in V} x_{ij}^k = 1 \qquad (i,j) \in V^2 \tag{5b}$$

$$\sum_{a \in A} g_{ij}^m \pi(a, a') \geq \sum_{k \in V} g_{ij}^k(a') x_{ij}^k \qquad m \in V, a' \in A \tag{5c}$$

$$\sum_{a \in A} c_a \pi(a, a') \leq c_a \Theta \qquad a' \in A \tag{5d}$$

$$x_{ij}^k \geq 0 \qquad (i,j,k) \in V^3 \tag{5e}$$

$$pi(a, a') \geq 0 \qquad a, a' \in A \tag{5f}$$

Constraints (5c) and (5d) enforce that the routing is optimal for the worse case TM based on capacity constraints; see [5] for further details. One advantage of this formulation is that it defines a routing that does not need to be reoptimized through time as all possible TMs are implicitly considered. As a drawback, the size of the formulation is significantly larger because of variables $\pi(a, a')$ and constraints (5b).

We now present the formulations used in the online procedures presented in Section IV. Based on formulation SR-1 and a set of TMs $\mathcal{D}$, formulations R-SR-w and S-SR-w used for the online procedures RO and SO can be adapted for a single waypoint. The robust routing R-SR-1 is defined as follows:

(R-SR-1)

$$\min \quad \Theta \tag{6a}$$

$$\text{s.t.} \quad \sum_{k \in V} x_{ij}^k = 1 \qquad (i,j) \in V^2 \tag{6b}$$

$$\sum_{(i,j) \in V^2} d_{ij}^D \sum_{k \in V} g_{ij}^k(a) x_{ij}^k \leq c_a \Theta \qquad a \in A, D \in \mathcal{D} \tag{6c}$$

$$x_{ij}^k \geq 0 \qquad (i,j,k) \in V^3 \tag{6d}$$

As before, equations (6b) ensure that the demands are fully routed, while inequalities (6c) combined with the minimizing objective function force the MLU Θ to be equal to the MLU of worst TM in D.

The stochastic routing S-SR-1 is defined as follows:

(S-SR-1)

$$\min \quad z + \frac{1}{1-\epsilon} \sum_{D \in \mathcal{D}} p^D y^D \tag{7a}$$

$$\text{s.t.} \quad \sum_{k \in V} x_{ij}^k = 1 \qquad (i,j) \in V^2 \tag{7b}$$

$$\sum_{(i,j) \in V^2} d_{ij}^D \sum_{k \in V} g_{ij}^k(a) x_{ij}^k \leq c_a \Theta^D \qquad a \in A, D \in \mathcal{D} \tag{7c}$$

$$y^D \geq \Theta^D - z \qquad D \in \mathcal{D} \tag{7d}$$

$$y^D \geq 0 \qquad D \in \mathcal{D} \tag{7e}$$

$$x_{ij}^k \geq 0 \qquad (i,j,k) \in V^3 \tag{7f}$$

Inequalities (7c) fix Θ^D to the MLU for the TM $D \in \mathcal{D}$ while (7d) and (7e), together with the objective function, enforce the proper computation of the CVaR.

VI. Numerical results

Numerical experiments are performed with an 8-core Intel Xeon CPU E5-2670 v3 at 2.30GHz with 32 Gb. The robust and stochastic optimization is modelled in Julia 1.7.2 and solved with Gurobi 9.5.1. The time limit when solving the routing problem is set to 3600 s. A value $\epsilon = 10\%$ is used for the CVaR, representing a moderate aversion to risk.

A. Instances

The online algorithms are tested on networks provided by REPETITA [22], a framework aimed at easing repeatable experiments on TE algorithms, with associated OSPF metrics. For each network, initial TMs are generated with a gravity model proposed by [23], which produces realistic TMs for internet networks. The final TMs used in the experiments are then generated by averaging n consecutive matrices from the initial set, with $n \in \{2, 5, 10\}$, leading to a maximum variation in demand between two nodes of 50%, 20% and 10% in two consecutive TMs. To evaluate the improvement of SR on OSPF easily, all TM are normalized so that the MLU is equal to 1 when the routing is performed with OSPF only.

B. Online vs. oblivious routing

The performance of the online procedures is compared to the oblivious routing obtained with O-SR-1, proposed by Bhatia et al. [5]. We compare our online procedures with the model of Bhatia et al. on networks solved under 3600 s with O-SR-1.

Our tests were performed over three sets of 100 consecutive TMs with a maximum variation in demand of respectively 50%, 20%, and 10%. Table I reports the solving time of the model of Bhatia et al., the maximum solving time of the robust (R-SR-1) and stochastic (S-SR-1) formulations over all iterations of the online procedure, and the mean (μ) and standard deviations (σ) of the MLUs. The MLUs reported for the online procedure are those obtained before reoptimizing the routing based on the current TM. Column Imp indicates the percentage of TM for which the online routing provides a better MLU than the oblivious routing O-SR-2. Overall, both online procedures find better routings than O-SR-1, especially when the variation in demand between two consecutive TM is smaller. The lowest MLUs are obtained with the SO procedure, while the RO procedure has smaller solving times as the formulations consider a smaller number of TMs. The maximum number of TMs in the robust model is indicated in column Nb. TMs. The solving time of O-SR-1 is significantly larger than for R-SR-1 and S-SR-1 because of the size of the formulation. The oblivious approach is overly conservative and, as expected, performs poorly for many matrices. A possible explanation is that the model of Bhatia et al. [5] corresponds roughly to our robust approach with a set $\mathcal{D}$ containing all matrices that can be segment routed. Therefore, the focus is on very pathological TM cases unlikely to appear in a real setting. The RO procedure's worse results than the SO online procedure can be explained similarly as the pathological TMs are added to $\mathcal{D}$ and kept throughout the procedure.

The computation time of each iteration for network Sunet. is illustrated in Figure 4. The computation time of SO increases linearly as the number of TMs in the model increases. The RO curve illustrates that the routing is reoptimized only when new worse-case TMs are found, limiting the increase in computation time. The SO-30 curve shows that a constant time is reached for the maximum number of TMs to consider (the 30 last TMs plus 1 central matrix averaged over the former 30 TMs).

Figures 5 and 6 illustrate the evolution of the MLU for networks BtEur and Hurri. The MLUs are reported pre-optimization (as reported in Table I) and post-optimization for both procedures. The minimum possible MLU is reported as Optimal SR and is obtained by optimizing the routing with SR-1 on the TM of the current iteration. As observed in Table I, both online procedures provide better results, especially the stochastic one. We also observe that the routing obtained post-optimization in the SO procedure is almost always as efficient as the Optimal SR. On the other hand, the RO procedure stays relatively far from the Optimal SR, confirming that a robust

Network	Var (%)	Bhatia et al.			RO					SO			
		t (s)	MLU (%)		t (s)	MLU (%)		Nb. TMs	Imp. (%)	t (s)	MLU (%)		Imp (%)
			μ	σ		μ	σ				μ	σ	
Highw.	50.0	635.92	0.92	0.11	2.86	0.98	0.09	33	34.0	15.14	0.9	0.14	66.0
$\|V\| = 18$	20.0	635.92	0.88	0.09	0.69	0.95	0.04	10	27.0	12.72	0.83	0.1	90.0
$\|A\| = 106$	10.0	635.92	0.87	0.07	0.66	0.95	0.04	10	14.0	11.58	0.8	0.07	94.0
Easyn.	50.0	28.97	1.07	0.1	3.51	0.83	0.08	43	99.0	19.8	0.78	0.09	99.0
$\|V\| = 19$	20.0	28.97	1.08	0.09	3.93	0.77	0.05	49	100.0	15.06	0.72	0.07	100.0
$\|A\| = 58$	10.0	28.97	1.08	0.06	3.2	0.72	0.04	42	100.0	15.04	0.69	0.05	100.0
Janet	50.0	33.21	1.02	0.04	6.53	1.12	0.18	63	52.0	18.45	1.11	0.16	47.0
$\|V\| = 20$	20.0	33.21	1.02	0.03	4.85	1.07	0.12	51	59.0	14.0	1.06	0.1	60.0
$\|A\| = 80$	10.0	33.21	1.02	0.02	4.22	1.04	0.07	45	70.0	12.81	1.03	0.05	70.0
BtEur.	50.0	337.08	0.99	0.09	9.31	1.03	0.1	57	52.0	37.83	0.97	0.1	78.0
$\|V\| = 24$	20.0	337.08	0.98	0.08	6.78	0.98	0.09	44	69.0	28.86	0.92	0.09	92.0
$\|A\| = 74$	10.0	337.08	0.98	0.06	5.71	0.97	0.06	38	65.0	28.88	0.87	0.09	90.0
Hurri.	50.0	3459.5	0.94	0.11	6.28	0.98	0.09	38	42.0	55.74	0.85	0.12	83.0
$\|V\| = 24$	20.0	3459.5	0.92	0.1	8.6	0.87	0.08	49	71.0	55.48	0.72	0.09	99.0
$\|A\| = 74$	10.0	3459.5	0.92	0.08	9.67	0.72	0.06	53	100.0	62.98	0.64	0.06	100.0
Sunet.	50.0	2167.13	1.1	0.05	12.96	1.02	0.08	40	90.0	58.73	1.0	0.1	96.0
$\|V\| = 26$	20.0	2167.13	1.11	0.04	7.08	1.0	0.04	26	100.0	39.29	0.99	0.05	97.0
$\|A\| = 98$	10.0	2167.13	1.11	0.02	3.46	1.0	0.01	16	100.0	37.77	0.99	0.03	99.0

TABLE I: Results from Bhatia et al. and the RO and SO procedures over 100 TMs

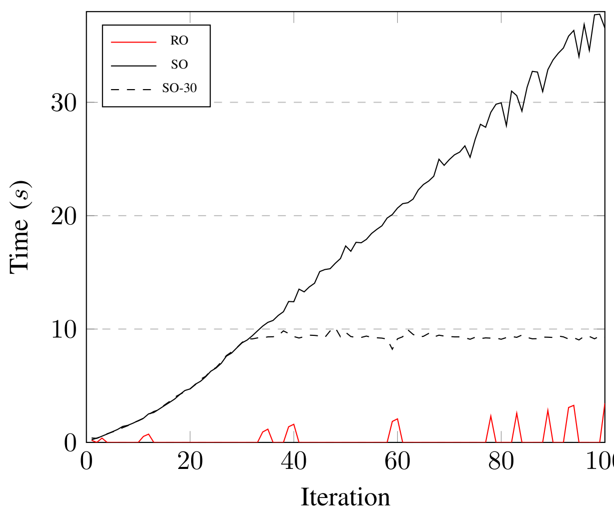


Fig. 4: Computation time per iteration on Sunet. ($|V| = 26, |A| = 98$), 10% variation in demand

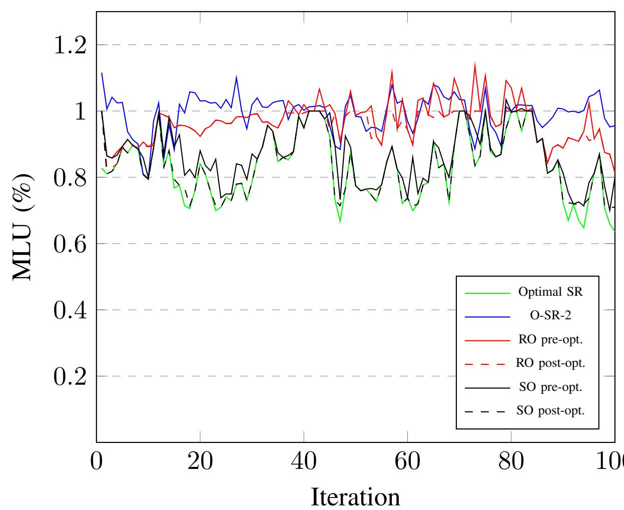


Fig. 5: MLU comparison on BtEur. ($|V| = 24, |A| = 74$) with 10% variation

procedure is too conservative.

Table II provides results of the SO procedure when considering a limited number of TMs. As seen in Figure 4, the maximum solving time per iteration significantly decreases when limiting the number of TMs. As one could expect, the average MLU increases as the variation in demand increases and the number of TMs decreases. Still, for TMs with a 10% variation, the MLU is, on average, only 1% worse when considering only 10 TMs in SO-10 rather than all of them in SO.

C. Larger networks

Results on larger networks are reported in Table III on 50 consecutive TMs. TMs with a variation of 10% between two consecutive ones are considered. The average MLU is reported pre- and post-optimization. A time-out and a run-out of memory are reported respectively as T.O. and MEM. along with the iteration at which the issue occurred in brackets. The average optimal MLU obtained with SR-1 considering the TMs are known in advance is reported in column Optimal MLU. The proportion of TMs added in RO increases significantly with the size of the network, all of them being added to $\mathcal{D}$ starting for Uunet and larger networks. This leads to memory issues and worse MLUs, especially pre-optimization. SO reaches the timeout on the largest network with about 100 nodes at the 9^{th} TM. The MLUs with SO-10 improve as the number of nodes of the network increases, which can be explained by the larger number of routing possibilities on large networks. We can also see the MLUs obtained with SO-10 is close to the optimal MLU, as illustrated in Figure 7, confirming the efficiency of the stochastic approach.

Network	Var (%)	SO-30				SO-20				SO-10			
		t (s)	MLU (%)		Imp. (%)	t (s)	MLU (%)		Imp. (%)	t (s)	MLU (%)		Imp.(%)
			μ	σ			μ	σ			μ	σ	
Highw.	50	3.45	0.89	0.12	75.0	2.0	0.91	0.13	64.0	0.91	0.93	0.12	54.0
$\lvert V\rvert = 18$	20	3.2	0.84	0.1	89.0	1.85	0.84	0.1	86.0	0.95	0.85	0.1	85.0
$\lvert A\rvert = 106$	10	3.39	0.81	0.07	95.0	1.91	0.81	0.07	95.0	0.9	0.81	0.07	92.0
Easyn.	50	3.81	0.79	0.09	100.0	2.3	0.79	0.09	100.0	0.89	0.81	0.1	99.0
$\lvert V\rvert = 19$	20	3.6	0.72	0.07	100.0	1.86	0.72	0.07	100.0	0.74	0.72	0.07	100.0
$\lvert A\rvert = 58$	10	3.42	0.69	0.05	100.0	1.85	0.69	0.05	100.0	0.84	0.7	0.05	100.0
Janet.	50	4.02	1.14	0.18	40.0	2.57	1.15	0.19	43.0	0.9	1.19	0.21	34.0
$\lvert V\rvert = 20$	20	3.59	1.07	0.11	53.0	2.28	1.09	0.12	49.0	0.95	1.1	0.12	36.0
$\lvert A\rvert = 80$	10	3.43	1.04	0.05	48.0	2.1	1.06	0.08	43.0	0.9	1.06	0.07	47.0
BtEur.	50	7.14	0.98	0.11	77.0	4.3	0.99	0.12	70.0	1.69	1.0	0.09	72.0
$\lvert V\rvert = 24$	20	9.31	0.93	0.08	89.0	5.21	0.93	0.08	86.0	1.52	0.94	0.08	88.0
$\lvert A\rvert = 74$	10	7.23	0.87	0.09	90.0	4.97	0.87	0.08	90.0	1.78	0.87	0.09	91.0
Hurri.	50	8.66	0.86	0.12	82.0	5.72	0.87	0.13	82.0	2.41	0.88	0.12	74.0
$\lvert V\rvert = 24$	20	11.2	0.73	0.09	99.0	6.33	0.74	0.09	99.0	2.77	0.74	0.08	99.0
$\lvert A\rvert = 74$	10	10.66	0.65	0.06	100.0	6.43	0.65	0.06	100.0	2.56	0.66	0.06	100.0
Sunet.	50	13.08	1.0	0.09	96.0	6.27	1.0	0.1	95.0	2.49	1.0	0.08	94.0
$\lvert V\rvert = 26$	20	9.81	0.99	0.04	99.0	5.61	0.99	0.04	100.0	2.28	1.0	0.04	98.0
$\lvert A\rvert = 98$	10	10.11	0.99	0.03	99.0	5.4	0.99	0.04	99.0	2.37	1.0	0.03	99.0

TABLE II: Impact of limiting the number of demands in the SO procedure over 100 TMs

Instance			RO				SO-10			Optimal MLU
Network	$\lvert V\rvert$	$\lvert A\rvert$	t (s)	MLU		Nb. TMs	t (s)	MLU		
				Pre-opt.	Post-opt.			Pre-opt.	Post-opt.	
Deuts	30	110	6.77	0.99	0.99	18	3.98	0.99	0.98	0.97
Ntt	32	432	6.75	1.01	1.0	16	4.83	1.01	1.0	1.0
China	42	132	2.96	0.78	0.78	5	9.53	0.77	0.76	0.75
Palme	45	140	59.84	0.9	0.9	34	15.15	0.79	0.77	0.78
Uunet	49	168	467.89	1.4	0.84	50	28.28	0.84	0.8	0.8
synth	50	276	686.67	1.54	0.44	50	318.65	0.45	0.39	0.38
rf396	79	294	702.22	1.3	0.57	50	231.02	0.58	0.54	0.54
rf175	87	322	MEM. (38)	-	-	37	991.32	0.51	0.48	0.47
synth	100	572	MEM. (7)	-	-	6	T.O. (9)	-	-	-

TABLE III: Results on large instances of the online procedure over 50 TMs

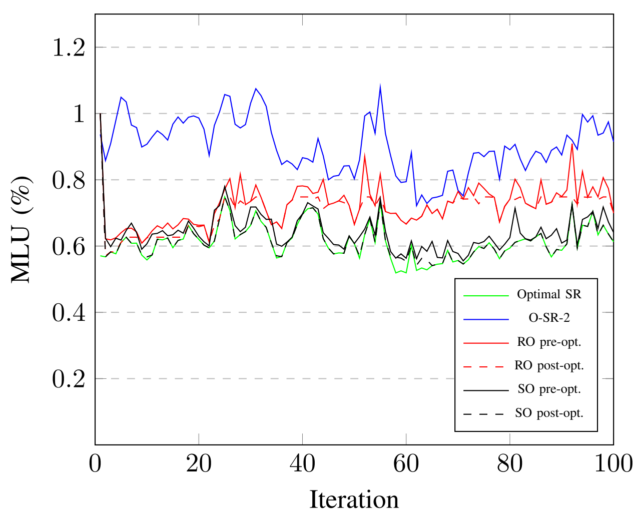


Fig. 6: MLU comparison on Hurri. ($|V| = 24, |A| = 74$) with 10% variation

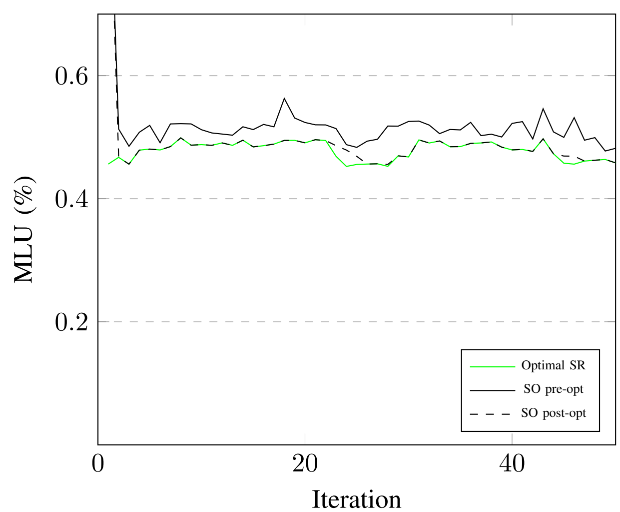


Fig. 7: MLU per iteration on rf175. ($|V| = 87, |A| = 322$) with 10% variation

VII. Conclusion

This paper studied TE in segment routing under demand uncertainty. We made a case for a stochastic approach and showed that it performs well across a wide range of scenarios. This approach outperforms robust approaches, which often turn out to be too conservative. While without further consideration, our stochastic approach has a higher runtime, we showed that it can be sped up significantly without sacrificing TE quality by reducing the number of scenarios and using only the recent traffic history observed. This allows to scale efficient segment routing to larger instances than with state-of-the-art methods. We also empirically studied the impact of demand volatility on the relative benefits of segment routing over traditional approaches.

The MLU and a single waypoint are considered in the numerical results of this paper for comparison purposes. Still, our methodology stays valid for any additive convex function in the loads on the arcs or any number of waypoints. Other deterministic SR formulations considering more waypoints can be embedded in the robust and stochastic models to be used in the online procedures proposed.

We understand our work as a first step and believe it opens several interesting avenues for future research. In particular, it would be interesting to explore whether the conditional value at risk approaches could also be useful in other contexts, such as datacenter networks, and how it could be combined with existing concepts such as semi-oblivious routing [24].

Acknowledgments

Research supported by the Austrian Science Fund (FWF), I 4800-N (ADVISE).

References

[1] M. Parham, T. Fenz, N. Suess, K.-T. Foerster, and S. Schmid, "Traffic engineering with joint link weight and segment optimization," in *Proc. 17th ACM International Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, 2021.

[2] N. Feamster, J. Rexford, and E. Zegura, "The road to sdn: an intellectual history of programmable networks," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 2, pp. 87–98, 2014.

[3] F. Aubry, "Models and algorithms for network optimization with segment routing," Ph.D. dissertation, UCLouvain, 2020.

[4] P. L. Ventre, S. Salsano, M. Polverini, A. Cianfrani, A. Abdelsalam, C. Filsfils, P. Camarillo, and F. Clad, "Segment routing: A comprehensive survey of research activities, standardization efforts, and implementation results," *IEEE Commun. Surv. Tutorials*, vol. 23, no. 1, pp. 182–221, 2021.

[5] R. Bhatia, F. Hao, M. Kodialam, and T. Lakshman, "Optimized network traffic engineering using segment routing," *IEEE Conference on Computer Communications (INFOCOM)*, pp. 657–665, 2015.

[6] J. Zerwas, I. Poese, S. Schmid, and A. Blenk, "On the benefits of joint optimization of reconfigurable cdn-isp infrastructure," in *IEEE Transactions on Network and Service Management (TNSM)*, 2021.

[7] B. Frank, I. Poese, Y. Lin, G. Smaragdakis, A. Feldmann, B. Maggs, J. Rake, S. Uhlig, and R. Weber, "Pushing cdn-isp collaboration to the limit," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 3, pp. 34–44, 2013.

[8] X. Xiao, A. Hannan, B. Bailey, and L. M. Ni, "Traffic engineering with MPLS in the internet," *IEEE Netw.*, vol. 14, no. 2, pp. 28–33, 2000.

[9] S. Agarwal, M. S. Kodialam, and T. V. Lakshman, "Traffic engineering in software defined networks," in *INFOCOM*. IEEE, 2013, pp. 2211–2219.

[10] C. Filsfils, S. Previdi, B. Decraene, S. Litkowski, and R. Shakir, "Segment routing architecture," in *IETF Internet-Draft*, 2017.

[11] B. Fortz and M. Thorup, "Increasing Internet Capacity Using Local Search," *Computational Optimization and Applications*, vol. 29, no. 1, pp. 13–48, 2004.

[12] M. Chiesa, G. Kindler, and M. Schapira, "Traffic engineering with equal-cost-multipath: An algorithmic perspective," *IEEE/ACM Transactions on Networking*, vol. 25, no. 2, pp. 779–792, 2017.

[13] M. Pióro, A. Szentesi, J. Harmatos, A. Jüttner, P. Gajowniczek, and S. Kozdrowski, "On open shortest path first related network optimisation problems," *Perform. Evaluation*, vol. 48, no. 1/4, pp. 201–223, 2002. [Online]. Available: https://doi.org/10.1016/S0166-5316(02)00036-6

[14] R. Hartert, "Fast and scalable optimization for segment routing," Ph.D. dissertation, UCLouvain, 2018.

[15] E. Moreno, A. Beghelli, and F. Cugini, "Traffic engineering in segment routing networks," *Computer Networks*, vol. 114, pp. 23–31, 2017.

[16] R. Bhatia, F. Hao, M. S. Kodialam, and T. V. Lakshman, "Optimized network traffic engineering using segment routing," in *INFOCOM*. IEEE, 2015, pp. 657–665.

[17] F. Aubry, S. Vissicchio, O. Bonaventure, and Y. Deville, "Robustly disjoint paths with segment routing," in *CoNEXT*. ACM, 2018, pp. 204–216.

[18] S. Balon, F. Skivée, and G. Leduc, "How well do traffic engineering objective functions meet te requirements?" *Lecture Notes in Computer Science*, vol. 3976, pp. 75–86, 2006.

[19] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *Journal of Risk*, vol. 2, pp. 21–41, 2000.

[20] C. Filippi, G. Guastaroba, and M. Speranza, "Conditional value-at-risk beyond finance: a survey," *International Transactions in Operational Research*, vol. 27, no. 3, pp. 1277–1319, 2020.

[21] E. Pujol, I. Poese, J. Zerwas, G. Smaragdakis, and A. Feldmann, "Steering hyper-giants' traffic at scale," in *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, 2019, pp. 82–95.

[22] S. Gay, P. Schaus, and S. Vissicchio, "Repetita: Repeatable experiments for performance evaluation of traffic-engineering algorithms," *arXiv:1710.08665v1*, 2017.

[23] M. Roughan, "Simplifying the synthesis of internet traffic matrices," *ACM SIGCOMM Computer Communication Review*, vol. 35, pp. 93–96, 2005.

[24] P. Kumar, Y. Yuan, C. Yu, N. Foster, R. Kleinberg, P. Lapukhov, C. L. Lim, and R. Soulé, "Semi-oblivious traffic engineering: The road not taken," in *15th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 18)*, 2018, pp. 157–170.