Definition and evaluation of a strategy for Real-Time Traffic Engineering in SD-WAN

Michele Beccari - 856608

1 Introduction

Wide Area Networks (WANs) are adopted by large companies and organizations when connecting their branches to overcome the potential reliability issues that may arise by just using general connectivity (i.e the standard connections offered by Internet Service Providers (ISPs), usually sold for personal use). However, while offering greater guarantees, setting up a full WAN connectivity is massively more expensive than using ISPs; some WANs may even require renting actual cables from ISPs. SD-WANs (Software Defined WAN) offer a hybrid approach to find a middle ground between the reliability of standard WANs and the economic advantage of general connectivity.
SD-WAN work by stipulating multiple contracts of generalized internet connections (e.g 3G, 4G, PON,etc.) and then combining the different links to guarantee a certain quality of service level. Some of the more traditional and more expensive WAN solutions may still be used (such as using a different network, completely detached from general connectivity), but their usage could be reduced to handle just the applications that strictly require high level of reliability. Organizations adopting SD-WAN solutions usually need to route traffic coming from applications having varying requirements in term of required delay, bandwidth, error rate, etc. These technical requirements are usually formalized by Service Level Agreements (SLA).

2 The Problem

When using both standard WAN connectivity solutions and general connectivity traffic may be routed through the different links according to the requirements of the source of the traffic itself. (i.e., the more expensive links are only used when strictly necessary and doing otherwise may cause SLA violations) This thesis tries to explore a strategy to route traffic across different kinds of links to minimize the overall transmission cost, while still respecting the SLA requirements of each source.

3 Proposed strategy

The basic approach to solve the problem is to use a linear optimization strategy, such as in the *Intent-Based Routing Policy Optimization in SD-WAN* paper [1].
The proposed scenario is composed by:

- A set of applications with varying SLA requirements that generate data periodically (in our case a sinusoidal pattern) for a pre-determined amount of time
- A set of links with different characteristics in term of bandwidth, delay, error rate, etc.

The linear optimization strategy is run periodically, to account for the new data generated by the applications; one of the challenges of the strategy is finding a way to account for the potential overuse of the cheaper links.
In this thesis we rely on a basic delay estimation function to try to guess the future load on a given link.

3.1 The linear optimization problem

The core of the strategy is the following linear optimization problem:

$$\min \quad \sum_{l \in L} c_l \cdot LU_l \tag{1}$$

$$\text{s.t.} \quad LU_l = \sum_{a \in A} t_a x_l^a \leq C_l, \qquad \forall l \in L, \tag{2}$$

$$f_l^a(x) \leq D_a, \qquad \forall a \in A, \forall l \in L, \tag{3}$$

$$D_a \leq \bar{D}_a, \qquad \forall a \in A, \tag{4}$$

$$e_l^a(x) \leq E_a, \qquad \forall a \in A, \forall l \in L, \tag{5}$$

$$E_a \leq \bar{E}_a, \qquad \forall a \in A, \tag{6}$$

$$b_l^a(x) \leq B_a, \qquad \forall a \in A, \forall l \in L, \tag{7}$$

$$B_a \leq \bar{B}_a, \qquad \forall a \in A, \tag{8}$$

$$\sum_{l \in L} x_l^a = 1, \qquad \forall a \in A \tag{9}$$

The variables are:

Variable	Range	Notes
c	R	The cost coefficient for a link
x_l^a	$[0, 1]$	The split ratio of each application $a \in A$ over each link $l \in L$
LU_l	$[0, 1]$	The utilization in percentage for each link $l \in L$
D_a	R	The maximum delay estimated by application a over all overlay link
$f_l^a(x)$	R	The delay function that provides the delay of application a over overlay link l considering the assignment given by x.
$e_l^a(x)$	R	The error rate function that provides the error rate of application a over overlay link l considering the assignment given by x.
$b_l^a(x)$	R	The bandwidth function that provides the required bandwidth rate of application a over overlay link l considering the assignment given by x.

The constraints are:

Constraint	Meaning
(2)	Ensures that the capacity of each link is satisfied
(3)	Computes the delay for each application over each link
(4)	Verifies the satisfaction of SLA delay requirements
(5)	Computes the error rate for each application over each link
(6)	Verifies the satisfaction of SLA error rate requirements
(7)	Computes the bandwidth usage for each application over each link
(8)	Verifies the satisfaction of SLA bandwidth requirements
(9)	Ensures that all pending packets are sent

4 Comparison with baseline strategies

The strategy has been tested and compared with three other strategies:

- Round robin strategy.
  This strategy will manage the packets generated by the different application by sending a packet to each link, looping until there are no more packages.
- Random strategy.
  This strategy will send any incoming packet to a randomly chosen link.
- Static Strategy
  This strategy will use a modified version of the main strategy.
  This modified version will not account for any delay caused by the excessive use of a given link. While unrealistic this serves as a lower bound.

5 Implementing and simulating the strategy

The strategy is implemented with a C++ custom simulator build on top of the *NS3* [2] framework. NS3 is a discrete-event simulator written with the C++ programming language which allows for easy simulation of networks.

At a very high level the code for the simulation is:

Algorithm 1 The simulation algorithm

while $ShouldBeRunning$ **do**
  $Result \leftarrow executeStrategy(A, L)$
  $enqueuePackages(Result)$
  if there are no longer pending packages and all applications have stopped **then**
    $ShouldBeRunning \leftarrow false$
  end if
end while

When running the experiments, the applications will stop after having generated a pre-determined amount of traffic. The applications will run (and therefore generate traffic) for a set amount of time and the simulation will stop when all the packets will have been sent.

5.1 Testing the strategy

The strategy is tested using a varying set of applications and a pre-determined set of interfaces. Varying amount of noise and shift will be applied to the traffic sinusoids to observe the changes in the overall cost of the transmission.

6 Test results

The test results shows how our strategy is able to outperform the other strategies, in both cost and fulfillment of the SLA requirements.

7 Future work

The strategy could be improved by providing better ways to estimate the future delay of sending data towards a given interface (e.g using an ML model to estimate the future traffic), and could be expanded to allow for user-defined function to also estimate future error rate and bandwidth availability.

References

[1] Pham Tran Anh Quang et al. Intent-based routing policy optimization in sd-wan. 2022.

[2] Ns3. https://www.nsnam.org/, last accessed on 04/03/2025.