Transport Assistants to Enhance TCP Connections: Investigating the Placement Problem

Jaime Galán-Jiménez[1], Mohamed Faten Zhani[2], José Gómez-delaHiz[1], and John Kaippallimalil[3]

[1]Universidad de Extremadura Escuela Politecnica
[2]Universite de Sousse Institut Superieur d'Informatique et des Technologies de Communication de Hammam Sousse
[3]Futurewei Technologies Inc

August 28, 2024

Abstract

As of today, TCP remains the de-facto transport protocol in the Internet. However, TCP may incur high delays, especially when retransmitting lost packets as they have to be retransmitted only by the source and after a timeout that is roughly equal to a round trip time. To reduce such delay, recent work [1–3] proposed to deploy a special network function, called Transport Assistant (TA), that is able to detect and retransmit lost TCP packets from inside the network rather than the source, and thereby, reduces retransmission delays. Unfortunately, there is no study on the impact of the placement of the TA on its performance benefits in terms of packet delivery delay. In this paper, we focus on the TA placement problem. We discuss the trade-offs and parameters to be considered to select the best placement for the TA. We first mathematically model the TCP packet delivery delay, i.e., the time needed to deliver TCP packets, when the TA is deployed. We also formulate, as an Integer Linear Program (ILP), the problem of placing multiple TAs in order to reduce TCP packet delivery delays while minimizing their deployment costs. We consider use-cases, one where a TA could handle a single flow and another where a TA could handle multiple flows. We then propose two heuristics to solve the problem with minimal execution time. Through experiments, we demonstrated that the deployment of TAs could reduce TCP packet delivery delays by up to 30% and could be leveraged to guide routing and load balancing. Moreover, we show that using the proposed heuristics for placing TAs could lead to performance that is close to optimal solutions obtained with the ILP but with lower execution time.

Transport Assistants to Enhance TCP Connections: Investigating the Placement Problem

Jaime Galán-Jiménez[1*], Mohamed Faten Zhani[2†],
José Gómez-delaHiz[1†], John Kaippallimalil[3†]

[1*]Department of Computer Systems and Telematics Engineering,
University of Extremadura, Avda. de la Universidad, S/N, Cáceres,
10003, Extremadura, Spain.
[2]ISITCom, University of Sousse, Sousse, 4011, Tunisia.
[3]Futurewei Technologies, Inc., USA.

*Corresponding author(s). E-mail(s): jaime@unex.es;
Contributing authors: mf.zhani@isitc.u-sousse.tn; jagomezdh@unex.es;
john.kaippallimalil@futurewei.com;
†These authors contributed equally to this work.

Abstract

As of today, TCP remains the de-facto transport protocol in the Internet. However, TCP may incur high delays, especially when retransmitting lost packets as they have to be retransmitted only by the source and after a timeout that is roughly equal to a round trip time. To reduce such delay, recent work [1–3] proposed to deploy a special network function, called Transport Assistant (TA), that is able to detect and retransmit lost TCP packets from inside the network rather than the source, and thereby, reduces retransmission delays. Unfortunately, there is no study on the impact of the placement of the TA on its performance benefits in terms of packet delivery delay. In this paper, we focus on the TA placement problem. We discuss the trade-offs and parameters to be considered to select the best placement for the TA. We first mathematically model the TCP packet delivery delay, i.e., the time needed to deliver TCP packets, when the TA is deployed. We also formulate, as an Integer Linear Program (ILP), the problem of placing multiple TAs in order to reduce TCP packet delivery delays while minimizing their deployment costs. We consider use-cases, one where a TA could handle a single flow and another where a TA could handle multiple flows. We then propose two heuristics to solve the problem with minimal execution time. Through experiments, we demonstrated that the deployment of TAs could reduce TCP packet delivery delays by up to 30% and could be leveraged to guide routing and load balancing. Moreover, we show that using the proposed heuristics for placing TAs could lead to performance that is close to optimal solutions obtained with the ILP but with lower execution time.

Keywords: Transport Control Protocol, Transport Assistant, Software-Defined Networks, FlexNGIA

1 Introduction

Since the genesis of Internet, the Transport Control Protocol (TCP) has been the main transport protocol used to support applications requiring performance and reliability. TCP provides a connection-oriented communication service over the Internet and offers a reliable transport between two endpoints over a non-reliable infrastructure supported by the Internet Protocol (IP) [4]. The TCP protocol incorporates several functions like flow control, congestion control, and packet loss detection and retransmission.

Thanks to the use of timers and duplicate Acknowledgments (ACKs), TCP is able to detect when a segment is lost and acts accordingly by retransmitting it. Unfortunately, this scheme incurs high delays to deliver packets that have experienced loss between the two endpoints. Indeed, the sender should wait for a timeout (roughly equal to a round trip time) before retransmitting the lost packet. This leads to a packet delivery delay that is at least three times the source-to-destination end-to-end delay. Of course, this high delay could severely affect the performance of several time-sensitive applications.

A potential solution to address this problem and reduce such delays could be to retransmit the lost segment from an intermediate node belonging to the path connecting the source to the destination. This solution is partially studied in [1–3] where the authors propose to deploy a special network function inside the network named *Transport Assistant* (TA). The TA is a network function that is deployed in an intermediate node and that is able to cache packets, to detect losses and retransmit lost packets. As the TA is closer to the destination, it is able to early detect and retransmit lost segments and to reduce the retransmission delays, avoiding the need to resort to the source node to retransmit the segment. As described in [3], the TA caches and analyzes in real-time incoming TCP segments in both directions. If a segment is not acknowledged within a timeout or if three duplicate acknowledgement are received requesting that segment, the TA restransmits it without waiting for the source retransmission. Whenever a segment is acknowledged, it is removed from the cache.

The experiments conducted in [2, 3] show that the use of TA clearly improves several performance metrics like the average packet delivery time, the flow completion time, the packet loss and the number of retransmitted packets from the source. However, selecting the best placement of the TA in the network remains a key challenge that has not been investigated in previous work. Indeed, the location of the TA has a direct impact on the performance and efficiency of the TA as it affects the time needed to detect loss as well as the retransmission delays.

In this work, we aim at addressing this particular challenge and focus on the TA placement problem to analyze the different parameters that could impact the performance benefits brought by the TA and could be used to optimally select its placement in the network.

This paper extends our previous work [5] where we propose a mathematical formula to model the expected packet delivery delay i.e., the average time needed to deliver a TCP packet assuming the TA is placed in the path of the flow and that packets could be lost and retransmitted multiple times. This time includes potential retransmission times and is computed in function of the location of the TA as well as parameters like the packet loss probabilities and propagation delays of the network links. We hence studied the impact of these parameters for different realistic scenarios and network topologies.

This paper extends these efforts by proposing the following contributions:

- We further extend the analysis of average packet delivery delay and the selection of the TA optimal placement in a more realistic scenario where the loss probability of the network links is variable over time.
- We also discuss how the use of TAs could improve routing decisions and load balancing and increase available bandwidth while satisfying delay constraints.
- Furthermore, we address the problem of the placement of multiple TAs in order to support multiple flows[1] and ensure their delay requirements. We propose two Integer Linear Programs (ILP1 and ILP2) aiming at finding the optimal number and location of multiple TAs in order to minimize their deployments costs and satisfy the delay requirements of the flows. In ILP1, a TA could only handle one flow whereas in ILP2, a TA could handle multiple flows at the same time. We also propose and evaluate two heuristic algorithms to solve the two programs for large-scale instances of the problem.

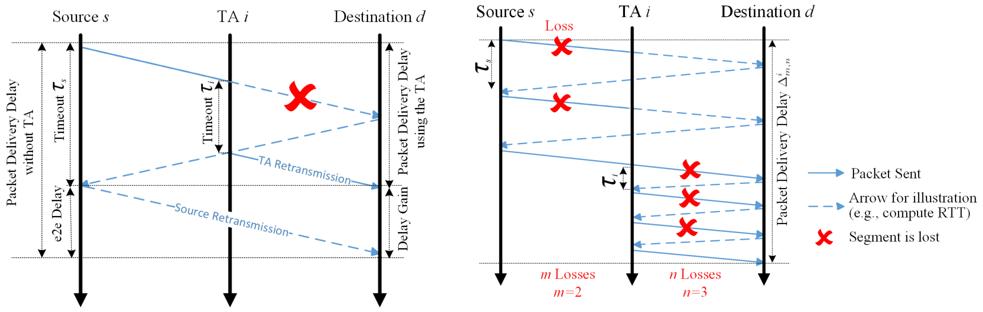
The rest of the paper is structured as follows. Section 2 addresses the TA placement problem for a single flow and presents the proposed mathematical model of the expected TCP packet delivery delay when the TA is used, including the impact of different parameters on this delay and how the TA could be leveraged to improve routing and load balancing decisions. Section 3 focuses on multi-TA placement problem for a single or multiple flows and presents the proposed ILPs, heuristics as well as their performance evaluation. Section 4 surveys related work, and finally, Section 5 concludes the work.

2 TA Placement for a Single Flow

In this section, we focus on the TA placement problem assuming that a *Transport Assistant* can handle only one single flow. We first define and mathematically model the expected packet delivery delay in function of parameters like the location of the TA and the loss probabilities and the propagation delays of the network links. Using the proposed formula, we study the impact of these parameters on the performance and

---

[1] In this paper, a flow refers to a set of TCP connections having the same source and destination and sharing the same path.



(a) Packet delivery delay with and without Transport Assistant

(b) Packet delivery assuming the segment is lost m times between the source s and the TA (node i) and lost n times between the TA and the destination d

Fig. 1: Packet delivery delay

efficiency of the TA. In particular, we first study the case where loss probabilities of the network links are constant over time. We then study the case where loss probabilities could vary over time. The results will show how loss probability and location could significantly affect packet delivery time, and how the location of the TA should also be adjusted based on the current congestion status of the network. Finally, we discuss how TAs could be leveraged to further support routing and load balancing decisions.

2.1 Mathematical Modeling of the Packet Delivery Delay

In order to evaluate the impact of the TA location on the performance, we define two key metrics. The first metric is called the reachability of the TA. It refers to the probability that a packet sent from the source reaches successfully the TA. The second metric is expected/average packet delivery time assuming a particular placement of the TA and given loss probabilities inside the network.

- *Reachability:* we define the reachability of the TA as the probability that a packet sent from the source reaches the Transport Assistant. Indeed, if the packet is lost before reaching the TA, it cannot be retransmitted from the TA, and hence, it will not benefit from the advantages of the TA. The reachability depends on the packet loss of the different links composing the path connecting the source to the location of the TA. In the following, we develop a formula to compute the reachability of the TA for a given location.

Let's assume that V is the set of nodes composing the network. Assume $\mathbb{P} = (v_1, v_2, ..., v_d)$ to be the path followed by the flow packets when traveling from the source to the destination. Hence, $(v_j)_{j\in[1,d]} \subset V$ is the subset of nodes composing the path originating from the source $v_1 \in \mathbb{P}$ towards the destination $v_d \in \mathbb{P}$. Furthermore, let $p_{j,j+1}$ be the packet loss probability in the link (v_j, v_{j+1}) where $j \in [1, d-1]$. We denote by $r_{u,w}$ the probability that a packet sent from node v_u reaches the node

v_w $(1 \leq u < w \leq d)$. This probability could be computed as follows:

$$r_{u,w} = \prod_{j=u}^{w-1} (1 - p_{j,j+1}) \tag{1}$$

Assuming that the node $v_i \in \mathbb{P}$ is the node hosting the TA, the reachability to the TA from the source could be given by $r_{1,i}$ computed using Eq.(1).

- *Packet Delivery Delay:* the packet delivery delay refers to the time needed for a packet to be delivered from the source to the destination. This time encompasses all delays including the time needed to retransmit the packet when it has been lost. In the following, we develop a formula to compute this time assuming a packet has been lost m times between the source and the TA and n times between the TA and the destination.

As shown in Fig. 12a, let τ_1 be the retransmission timeout used by the source v_1 to detect packet loss, and τ_i the one used by the TA located in node v_i to detect packet loss. These timeouts could be roughly estimated as the round trip times between the source or the TA and the destination, respectively. Hence, they could be estimated as follows:

$$\tau_1 = \delta_{1,d} + \bar{\delta}_{d,1} \tag{2}$$

$$\tau_i = \delta_{i,d} + \bar{\delta}_{d,i} \tag{3}$$

where $\delta_{v,w}$ is the delay between nodes v and w, and $\bar{\delta}_{w,v}$ is the delay of the return path from nodes w to v. Fig. 12a compares packet delivery times with and without the TA and shows clearly the different the used timeouts and the potential gain of using the Transport Assistant in terms of packet delivery time.

Let $\Delta^i_{m,n}$ be the packet delivery time of a segment assuming this segment has been lost m times between the source v_1 and the TA located at v_i and also assuming the same packet is lost n times between v_i and the destination v_d. Fig. 12b illustrates how such delay could be computed. Mathematically, it could be written as follows:

$$\begin{aligned} \Delta^i_{m,n} &= m\tau_1 + \delta_{1,i} + n\tau_i + \delta_{i,d} \\ &= m(\delta_{1,d} + \bar{\delta}_{d,1}) + \delta_{1,i} + n(\delta_{i,d} + \bar{\delta}_{d,i}) + \delta_{i,d} \\ &= m(\delta_{1,d} + \bar{\delta}_{d,1}) + n(\delta_{i,d} + \bar{\delta}_{d,i}) + \delta_{1,d} \end{aligned} \tag{4}$$

It is worth noting that if no TA is deployed, $\Delta^d_{m,0}$ provides the packet delivery time assuming the packet has been lost m times. In other words, it is equivalent to having the TA placed at the destination (i.e., $i = d$ and $n = 0$).

- *Expected Packet Delivery Delay:* based on probability theory, the expected/average packet delivery delay assuming the TA is placed at node v_i could be estimated as follows:

$$E(\Delta^i) = \sum_{(m,n) \in \mathbb{N}^2} P^i(m,n) \Delta^i_{m,n} \tag{5}$$

where $P^i(m,n)$ is the probability that the segment is lost m times between the source v_1 and the TA (located at v_i) and lost n times between v_i and the destination v_d. It could be estimated as follows:

$$P^i(m,n) = \begin{cases} 0 \quad \text{if } r_{1,i} = 1 \text{ and } r_{i,d} = 1 \\ (1 - r_{1,i})^m \; r_{1,i} \; (1 - r_{i,d})^n \; r_{i,d}, \quad \text{otherwise} \end{cases} \tag{6}$$

Note also that $E(\Delta^d)$ provides the expected packet delivery time when no TA is deployed. In this case, the TA is assumed to be placed at the destination (i.e., $i = d$, $n = 0$ and $r_{d,d} = 0$).

- *Selection of the Optimal Placement of the TA:* in order to select the optimal placement of the TA, i.e., the node that provides the minimum packet delivery delay, we can compute the expected packet delivery delay, $E(\Delta^i)$, for all nodes $v_i \in \mathbb{P}$ and then identify the optimal location.

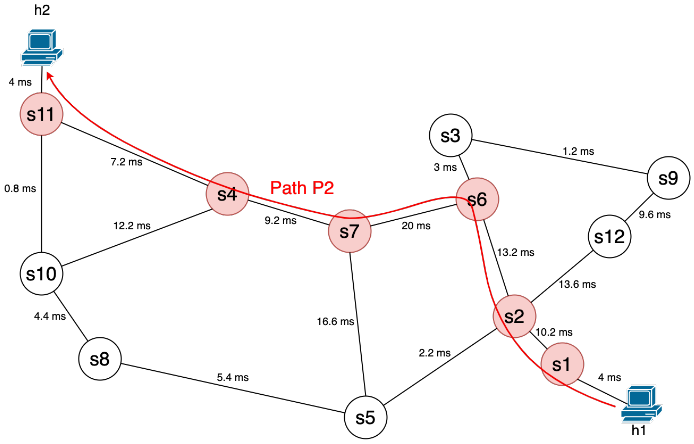


Fig. 2: Abilene topology. Links propagation delays follow a normal distribution with a mean of $8ms$ and standard deviation of $5ms$.

2.2 Analysis of the impact of the TA location, loss probabilities and propagation delays within the network

In the following, an exhaustive analysis of the impact of the TA location and the loss probabilities and propagation delays of the network's links is performed using the developed formula of the expected packet delivery time (Eq. 5). Experiments were conducted considering the Abilene network topology composed of $\mathcal{N} = 12$ nodes and $\mathcal{L} = 15$ bidirectional links (see Fig. 13). The figure shows the propagation delays of the all the links as well as the studied path (Path P2 in Table 3) on which a single TA should be placed to support TCP connections originating from host $h1$ towards

Table 1: Available paths from host h1 to host h2 in the Abilene Topology

Path ID	**Path**	**Number of nodes**	**Propagation delay (ms)**
$P1$	$(s1, s2, s5, s7, s4, s11)$	6	53.4
$P2$	$(s1, s2, s6, s7, s4, s11)$	6	67.8
$P3$	$(s1, s2, s5, s8, s10, s11)$	6	31.0
$P4$	$(s1, s2, s5, s7, s4, s10, s11)$	7	59.2
$P5$	$(s1, s2, s6, s7, s4, s10, s11)$	7	73.6
$P6$	$(s1, s2, s5, s8, s10, s4, s11)$	7	49.6
$P7$	$(s1, s2, s6, s7, s5, s8, s10, s11)$	8	78.6
$P8$	$(s1, s2, s12, s9, s3, s6, s7, s4, s11)$	9	82.0
$P9$	$(s1, s2, s6, s7, s5, s8, s10, s4, s11)$	9	97.5
$P10$	$(s1, s2, s12, s9, s3, s6, s7, s4, s10, s11)$	10	87.8
$P11$	$(s1, s2, s12, s9, s3, s6, s7, s5, s8, s10, s11)$	11	92.8
$P12$	$(s1, s2, s12, s9, s3, s6, s7, s5, s8, s10, s4, s11)$	12	111.4

host $h2$.

- **Results with constant loss probability over time:** The first set of results aims at evaluating the impact of the placement of the TA in the network in the studied path assuming that packet loss probabilities of the network links are constant over time. We consider first three scenarios 1, 2 and 3 (Table 4) where we keep the same propagation delays for the links but we consider different loss probabilities of the links for each of the three scenarios. In scenario 1, only access links have high loss probabilities (10%) whereas the other links have no packet loss. In scenarios 2, packet loss probabilities of the links are randomly set between 0 and 10%. For scenario 3, a different configuration of loss probabilities is generated. This is to show how the placement is impacted for different configurations.

Results for the three considered scenarios are reported in Fig. 14 where the Expected Packet Delivery Delay is shown as a function of the node in which the TA is placed. The figure shows also the Expected Packet Delivery Delay when no TA is used at all (see last bars in the figure). We can see in the figure that, for the three scenarios, the use of the TA allows to reduce the packet delivery delay. For instance, for scenario 1, the best location for the TA is the node s11 where the average packet delivery delay could be reduced up to 28% (from 100ms when the TA is not used to 82ms when it is deployed in s11). This is explained by the fact that retransmitting lost packets from s11 allows to avoid the delay needed to re-send the packet from h1 to s11. We can also see in the figure that, for scenarios 2 and 3, thanks to the TA, the packet delivery delay is reduced by up to 25% and 18% when the TA is deployed and placed

Table 2: Description of the studied scenarios. Note that all scenarios consider the studied path P2 (highlighted in Fig. 13)

Scenario	Propagation Delay	Packet Loss
1	Delays of P2 shown in Fig. 13	10% for access links (h1,s1) and (s11,h2) 0% for other links
2	Delays of P2 shown in Fig. 13	Random loss between 0% and 10% for all links
3	Delays of P2 shown in Fig. 13	Random loss between 0% and 10% for all links
4	Same in all links ($5ms$)	Random loss between 0% and 10% for all links
5	Long delay close to the source	Same as scenario 4
6	Long delay close to the destination	Same as scenario 4

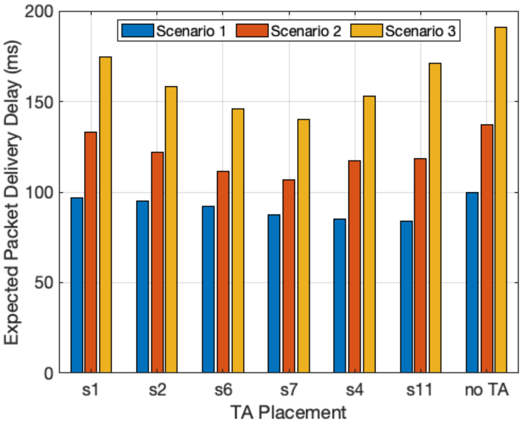


Fig. 3: Expected packet delivery delay for different TA placement in the path P2 for scenarios 1, 2 and 3 described in Table 4.

in nodes s7 and s6, respectively. It is also clear in the figure that the location of the TA has a strong impact on the resulting average packet delivery delay of the packets.

Furthermore, in order to study the impact of the propagation delays on the packet delivery delays, we consider three more scenarios 4, 5 and 6 (Table 4) where the propagation delays in the network were varied. In scenario 4, the propagation of the all links throughout the studied path are equal to 5ms. In this case, the average packet delivery delay (Eq. 5) will mainly depend on the loss probabilities of the links. As shown in Fig. 15, even if the propagation delays are equal, the expected packet delivery delay varies depending on the location of the TA. In scenario 5, the propagation delay of the links closer to the source are the highest. This means that if a packet crosses those links successfully, it makes sense to deploy a TA right after to avoid that retransmitted packets experience the long propagation delays. This is clear in Fig. 15, where in scenario 5, using the TA even at the first node of the path (s1) reduces the

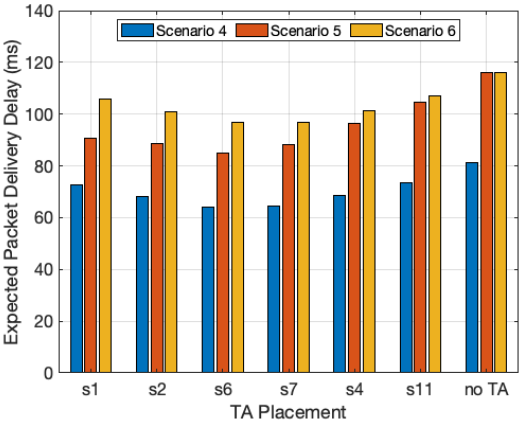


Fig. 4: Expected packet delivery delay for different TA placement in the path P2 for scenarios 4, 5, and 6 described in Table 4.

packet delivery delay (compared to the case where no TA is used shown in the last bar). Conversely, if the long links are towards the destination, i.e., scenario 6, the benefit of the TA becomes visible when it is farther from the source (e.g., when placed in node s7) and depends also on the loss probabilities of the links.

• **Results with variable loss probability over time:** In the next analysis, our goal is to evaluate the impact of a variable loss probability over time on the placement of the TA and the resulting performance in terms of expected packet delivery delay. In the following results, we use the same path P2 in Abilene Network. We assume that the packet loss probability of each link varies over time(see Fig. 16). As such the loss probability of each link, $p_{j,j+1}$, is randomly selected at each timeslot within the range $[0, 10\%]$. Fig. 17 shows packet loss probability over time for all the links of the considered path (path P2).

Results are reported in Fig. 17 where the expected packet delivery delay is computed over time based on the evolution of the packet loss probabilities of the links. The figure shows the results when the TA is not deployed and when it is deployed in different placements (i.e, nodes). We can see that the worst-case performance, i.e., highest packet delivery delay, occurs when the TA is not deployed. For this particular experiment and the packet loss evolution shown in Fig. 16, nodes s4 and s7 seem to be the best placements for the TA as they allow the TA to incur the lowest packet delivery delay. Accordingly, when the TA is used, it reduces up to 25% the packet delivery delay (see Fig. 17 from 5 to 10 timeslots).

The figure also shows that the optimal placement, i.e., the one achieving the lowest packet delivery delay, could change when the congestion state and loss probabilities in the network change over time. For instance, the best placement for the TA is s4 during the time interval [6,10] and it is the node s7 during the interval [10, 15].

2.3 TA-Assisted Routing and Load Balancing

As shown above, the deployment of the TA could significantly improve the packet delivery delay. To leverage such advantage, the network operator could use the proposed model to estimate the packet delivery delay for all available paths between a source and a destination assuming TAs are deployed. He can then deploy TAs and then route the flows across the paths satisfying the packet delivery delay requirements of these flows while ensuring the load is balanced across the selected paths.

For instance, Fig. 18 shows the expected packet delivery delay for all the possible paths between h1 and h2 (Table 3) in the Abilene topology (Fig. 13) without and with the TA deployed in the node that minimizes the packet delivery delay. For each path,

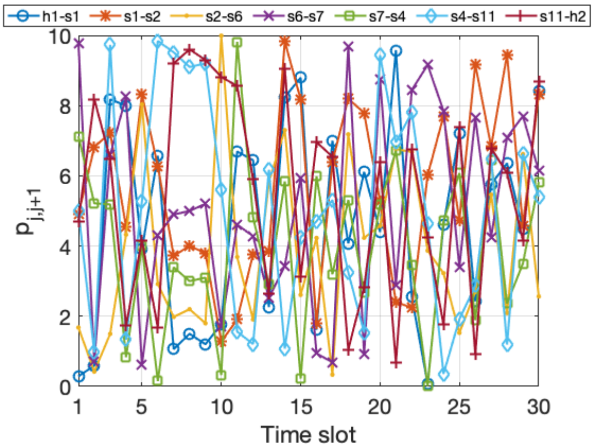


Fig. 5: Packet loss probability over time for all links composing the considered path P2.

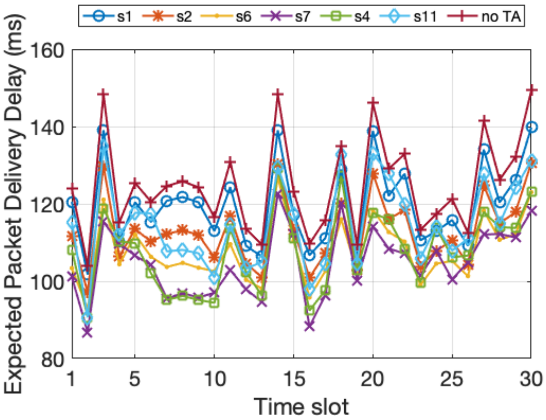


Fig. 6: Expected packet delivery delay over time for different TA placements.

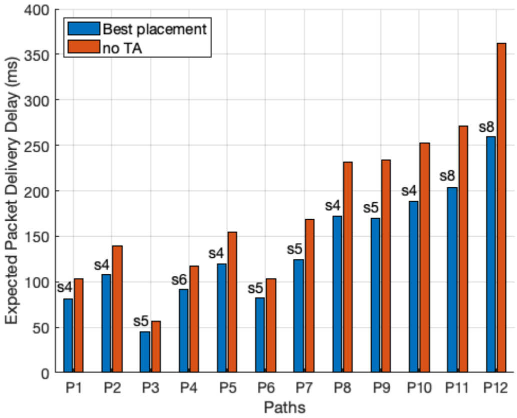


Fig. 7: Expected packet delivery delay for all paths between $h1$ and $h2$. Link propagation delays are the ones presented in Fig. 2.

the figure shows the best location (i.e., node) for the TA and the expected packet delivery delay. The figure clearly shows that the packet delivery delay could be reduced by up to 30% thanks to the TA (e.g., path P12).

Let's assume we have a set of flows between h1 and h2 having a requirement in terms of packet delivery delay of 200ms. Fig. 18 shows that without the TA, paths P1 to P7 satisfy this requirement and are available to route such flows. Thanks to the TA, three additional paths (P8, P9 and P10) could also satisfy the requirement and become available also to route such flows and to share the traffic load with the other paths satisfying the delay constraint. Fig. 19 shows similar results for the Abilene network but with different propagation delays for the links. The benefits of the TA in terms of packet delivery delays is also clear for all paths between h1 and h2.

These results confirm that the TA could be of utmost importance to reduce packet delivery delays of the available paths between the nodes, and hence, could be leveraged to increase the number of paths satisfying delay constraints and eventually improve traffic load balancing and increase the achievable bandwidth.

3 Multi-TA Placement for Multiple Flows

In this Section, we explore the multi-TA placement problem. Indeed, several Transport Assistants could be placed in the network to handle the flows of the network. In particular, a flow here could be defined as a set of TCP connections sharing the same source and destination. We assume that each flow has a delay constraint; that is the packet delivery delay for this flow should not exceed a predefined threshold. In this context, the goal of the *multi-TA placement problem* is to ensure all the flows of the network satisfy their delay constraints by placing a minimal number of TAs in the network while minimizing their deployment costs. The TAs are dynamically deployed when needed, i.e., in order to satisfy the delay constraint of the flow. Of couse, if

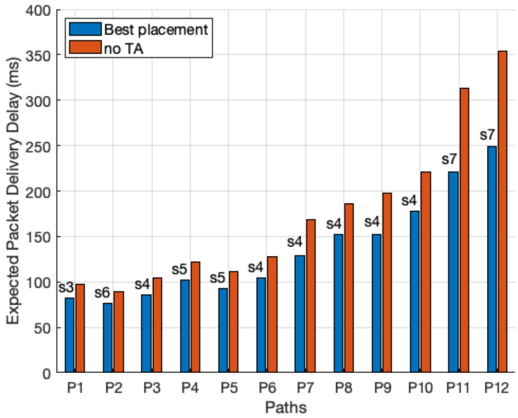


Fig. 8: Expected packet delivery delay for all paths between $h1$ and $h2$. Average link propagation delays equal to 8 ms.

the delay constraint of the flow is already satisfied there is no need for a Transport Assistant.

In the following, we propose two Integer Linear Programs (ILP1 and ILP2) aiming at finding the optimal number and location of multiple TAs in order to minimize their deployments costs and satisfy the delay requirements of the flows. In ILP1, a TA could only handle one flow whereas in ILP2, a TA could handle multiple flows at the same time. To address the problems for large-scale instances, We also propose and evaluate two heuristic algorithms.

3.1 Problem Formulation

In the following, we mathematically formulate the TA placement problem as a Integer Linear Program (ILP). Assume $\mathbb{F}$ is the set of all the flows crossing the network. To capture the delay constraint, for each flow $f \in \mathbb{F}$, we define λ_f as the maximum delay that should be experienced by the flow f.

Assuming $\mathbb{V} = (v_1, v_2, ..., v_d)$ is the set of the network nodes, we define a_i^f as a Boolean variable that is set to 1 if the flow f crosses the node v_i. Let c_i be the cost of deploying a TA in the node v_i. Of course, it could vary from one node to another depending on several parameters like the cost of the virtual machine or container hosting the TA, the bandwidth, and the processing capacity. We define b_f^i as a Boolean variable that is set to 0 if the node v_i is the destination of the flow f and 1 otherwise. This input variable is used to cancel the cost of the TA when it is not needed for the flow, i.e., the delay constraint of the flow is already satisfied. Thanks to b_f^i, we force the linear program to place the TA in the destination of the flow with a cost equal to zero. We denote by $E(\Delta_f^i)$ is the expected packet delivery delay for flow f assuming the TA is placed in node v_i. Finally, we define the decision variable x_f^i as a Boolean variable that takes 1 if the TA handling the flow f is placed in node v_i and 0 otherwise.

Our objective is to minimize the number and cost of the TA deployment over the infrastructure while ensuring the expected packet delivery delay of all the flows are below their respective maximum delay. In the following, we propose two ILPs (ILP1 and ILP2) and objective functions for two different cases. In the first formulation, ILP1, a single TA is assigned to only a single flow. In the second formulation, ILP2, a TA could handle several flows, and hence, it could be assigned to multiple TAs. In the following, we provide more details about the two proposed formulations.

- **ILP1 - One TA per Flow**: in this formulation, we assume that one TA could handle only the traffic of a single flow. In other words, there is a TA for each flow. The objective function aims at minimizing the costs of the deployments of the TAs. It can be expressed as follows:

$$\text{minimize} \sum_{f=1}^{|\mathbb{F}|} \sum_{i=1}^{|\mathbb{V}|} x_f^i b_f^i c_i \tag{7}$$

subject to the following constraints:

$$x_f^i \; E(\Delta_f^i) \leq \lambda_f \quad \forall i \in [1, |\mathbb{V}|] \quad \forall f \in [1, |\mathbb{F}|] \tag{8}$$

$$\sum_{i=1}^{|\mathbb{V}|} x_f^i = 1 \quad \forall f \in [1, |\mathbb{F}|] \tag{9}$$

$$x_f^i \leq a_i^f \quad \forall i \in [1, |\mathbb{V}|] \quad \forall f \in [1, |\mathbb{F}|] \tag{10}$$

Constraint (8) captures the delay requirement of a flow f and stipulates that the expected packet delivery delay when the TA is deployed is less or equal to the maximum delay that should be experienced by the flow λ_f. Constraint (9) ensures that only one single TA is deployed for each flow f. Finally, Constraint (10) ensures that a TA associated with one flow is deployed in a node crossed by that flow.

- **ILP2 - One TA for multiple Flow**: in this formulation, we assume that one TA could handle the traffic of multiple flows even if these flows have different sources and destinations.

Define y_i as a Boolean that takes 1 if a TA is deployed in node v_i. This means if a TA of multiple flows are placed in the same node v_i, they will be merged and a single TA will be instantiated in that node to handle these flows. Hence, y_i could be computed based on x_f^i as follows:

$$y_i \geq x_f^i b_f^i \quad \forall f \in [1, |\mathbb{F}|] \tag{11}$$

The objective function could be then written as follows:

$$\begin{aligned} &\text{minimize} \sum_{i=1}^{|\mathbb{V}|} y_i c_i \\ &\text{subject to constraints (8), (9) and (11)} \end{aligned} \tag{12}$$

3.2 Heuristic Algorithms

In order to tackle issues encountered in large-scale instances of the problem, (i.e., high number of nodes, paths and flows), we propose and assess in the following two heuristic algorithms to solve ILP1 and ILP2.

- **ILP1 - Heuristic 1:** in the following, a heuristic is proposed to solve ILP, i.e., the TA placement problem considering that one TA can handle the traffic of a single flow. Alg. 1 shows the pseudo-code of the proposed solution. Given as input the network topology, $\mathbb{G} = (\mathbb{V}, \mathbb{L})$, and the set of flows in the network, $\mathbb{F}$, the goal is to find a TA placement for each flow, while minimizing the associated deployment costs. Thus, for each flow $f \in \mathbb{F}$ (line 2), the shortest path rule is used to assess its corresponding *path*. For each node v_i in the path, we compute the expected packet delivery delay and evaluate the deployment cost c_i. The TA is placed in the nodes v_i that satisfies the flow delay constraint λ_f and having the lowest deployment cost (lines 6-11). Finally, the *flow_matrix* including the TA placement and the corresponding cost per flow is returned as output.

Algorithm 1 Heuristic to solve ILP1

```
Require: The network topology $\mathbb{G} = (\mathbb{V}, \mathbb{L})$, the set of flows: $\mathbb{F}$
 1: $flow\_matrix \leftarrow \emptyset$
 2: for all $f \in \mathbb{F}$ do                                  ▷ Handle each flow
 3:     $ta \leftarrow \emptyset$
 4:     $cost \leftarrow \infty$
 5:     $path \leftarrow \text{shortestPath}(\mathbb{V}, f_{src}, f_{dst})$      ▷ Shortest path for $f$
 6:     for all $v_i \in path$ do
 7:         if $E(\Delta^i) \leq \lambda_f$ and $c_i < cost$ then
 8:             $cost \leftarrow c_i$
 9:             $ta \leftarrow v_i$
10:         end if
11:     end for
12:     $flow\_matrix \leftarrow flow\_matrix \cup \{f, ta, cost\}$
13: end for
14: return $flow\_matrix$
```

- **ILP2 - Heuristic 2:** next, a new heuristic is proposed to solve the multi-TA placement problem, i.e., the TA placement problem considering that a single node can handle the traffic of multiple flows. The corresponding pseudo-code is reported in Alg. 2. In this case, the solution tries to exploit the set of nodes that already handled a TA for previous flows. Thus, for each flow f, the heuristic looks in the set of nodes composing the shortest path from the source to the destination if any of them handles a previous TA function. In case there is a node with a previous TA, the corresponding flow exploits this situation and no new TA is required (line 6). In case no node in the shortest path contains a TA, then the procedure to find the least cost

TA is adopted as in Alg. 1.

Algorithm 2 Heuristic to solve ILP2

```
Require: The network topology $\mathbb{G} = (\mathbb{V}, \mathbb{L})$, the set of flows: $\mathbb{F}$
flow_matrix ← $\emptyset$
for all $f \in \mathbb{F}$ do                                              ▷ Handle each flow
    ta ← $\emptyset$
    cost ← $\infty$
    path ← shortestPath($\mathbb{V}, f_{src}, f_{dst}$)                       ▷ Shortest path for $f$
    ta ← checkTAinPath(path)          ▷ Select the node in the path with a previous TA
    if ta == $\emptyset$ then
        for all $v_i \in$ path do
            if $E(\Delta^i) \leq \lambda_f$ and $c_i <$ cost then
                cost ← $c_i$
                ta ← $v_i$
            end if
        end for
    end if
    flow_matrix ← flow_matrix ∪ {$f$, ta, cost}
end for
return flow_matrix
```

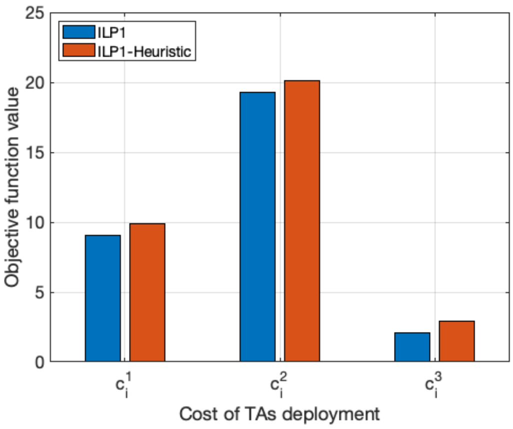


Fig. 9: Comparison between ILP1 and Heuristic 1 considering the three cost functions.

- **Experimental Results:** in this paragraph, we compare the performance of ILP1 vs. ILP1-Heuristic as well as the ILP2 vs. ILP2-Heuristic. To do so, we have run experiments on the Abilene Network (Fig. 13). We assume there is a flow between each pair of node, i.e., there is a total of $|\mathbb{F}| = |\mathbb{V}| \cdot (|\mathbb{V}| - 1)$ injected into the network.

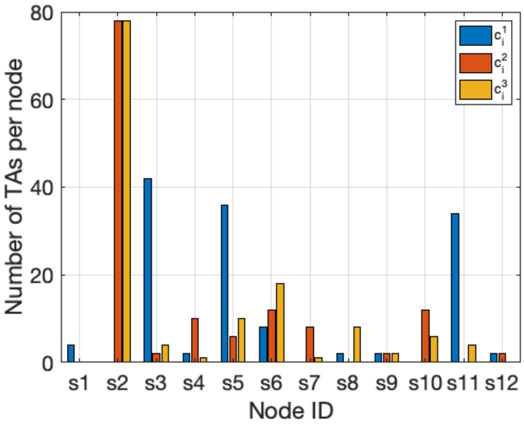


Fig. 10: Number of TAs per node for ILP1 considering the three cost functions.

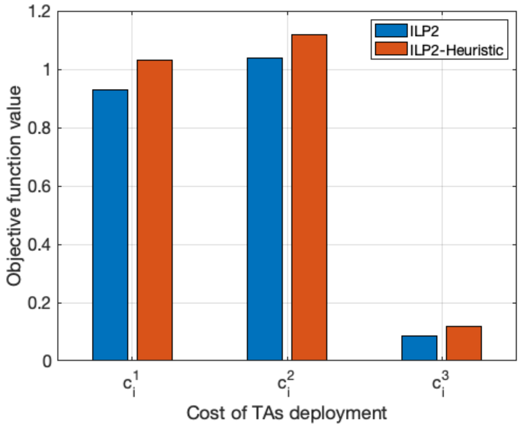


Fig. 11: Comparison between ILP2 and Heuristic 2 considering the three cost functions.

Each flow f has a random latency requirement $\lambda_f \in [min_i(E(\Delta_f^i)), max_i(E(\Delta_f^i))]$. We considered three different scenarios with different TA deployment cost functions c_i^1, c_i^2 and c_i^3 where $i \in \mathbb{V}$. For the first cost function, the cost of a node is randomly generated between 0 and 1 (i.e. $c_i^1 \in [0,1], \forall i \in \mathbb{V}$). In the second cost function, the deployment cost in node i is inversely proportional to its degree in the network graph, i.e., $c_i^2 = 1/degree(i)$. In the third cost function, c_i^3, the cost of the node is inversely proportional the number of flows crossing it. This gives lower costs to nodes that are used by the highest number of flows, and hence, gives them more priority to host TAs.

Fig. 20 shows a comparison of the objective function value found with the ILP1 and with Heuristic 1 for the three considered cost functions. It is clear from the figure that the heuristic provides a near optimal solution regardless of the considered cost function. Fig. 21 shows the number of TAs deployed in each node as found with ILP1 considering the three deployment cost functions. It shows the nodes where the TA could be provisioned taking into account the considered flows and the TA deployment costs.

Finally, Fig. 22 shows a comparison of the objective function value found with the ILP2 compared to Heuristic 2 for the three considered cost functions. Although the objective function with the heuristic algorithm is about 10% higher than that with ILP2. This shows that the heuristic provides a solution that is not far from the optimal one regardless of the considered cost function.

4 Related Work

In the last decades, a large body of work has been devoted to improve TCP and transport protocols performance in general [1, 3, 6–15]. For instance, Rosu et al., [12] advocated for splitting TCP connections between the client and server using proxy servers. They proposed a TCP Splicing service to accelerate the communication within the split-connection proxies by eliminating double copies of the packets between the buffers of the kernel and applications. Wang et al. [7] proposed to use edge switches to perform TCP-to-UDP conversion and to delegate packet retransmission from to intermediate switches. Wan et al. [10] attempted to minimize packet retransmission delays in wireless networks and devised a transport protocol using a hop-by-hop transmission where each hop could detect packet loss by detecting missing sequence numbers. Chen et al. [8, 9] designed a Transport protocol that leverages the control plane of software defined networks to establish transport connections and its data plane to ensure caching and retransmission.

Unlike these proposals requiring to replace TCP, the TA function proposed in [3] is designed to support TCP protocol without the need to alter it. This function is deployed within the network and could cache and retransmit TCP packets while TCP endpoints remain unaware of it. However, paper [3] does not discuss the placement problem and does not provide any mathematical model for the packet delivery delay incurred when the TA is used. We also highlight our previous work [16] that suggests to place the TA in the overloaded parts of the networks predicted thanks to Machine Learning techniques; However, it does not consider parameters like packet loss and propagation delays to identify the optimal placement of the TA and does not study their impact on packet delivery delays.

To our knowledge, this is the first study on the impact of the placement of the TA on the packet delivery delay using a mathematical model, which allows to quantify the TA benefits on the performance of TCP. This work is also the first to propose efficient TA placement algorithms taking into account packet delivery delay requirements.

5 Conclusion

In this paper, we focused on the TA placement problem and studied the parameters that could affect the performance of the TA. Hence, we developed a mathematical model that computes the expected packet delivery delay when the TA is deployed.

That is, the average time needed to deliver a TCP packet assuming the TA is operational and placed in the path of the flow. We also studied the impact of several parameters like the location of the TA as well as the packet loss probabilities and propagation delays of the network links for different realistic scenarios and network topologies. We also discuss how the deployment of multiple TAs could be leveraged for routing decisions and load balancing (i.e., TA-based routing).

Furthermore, we addressed the problem of multiple TA placement and proposed two Integer Linear Programs (ILP1 and ILP2) to find the optimal placement of the TAs to minimize their deployment costs and allow to satisfy the flow delay requirements in terms of maximal packet delivery delay. Two cases are considered where in the first (ILP1), a TA handles only one flow, and in the second (ILP2), a TA could handle multiple flows. We also proposed two heuristic algorithms to solve the two cases for large-scale instances of the problem and showed that the proposed heuristics provide solutions that are close to optimal solutions found by the ILPs.

As a future work, it would be interesting to propose network management schemes that could perform routing and TA placement in a joint manner in order to minimize packet delivery delays in the network. Indeed, Transport Assistants could be leveraged to satisfy service level agreements of flows with stringent performance requirements in terms packet delivery delays. Another interesting challenge would be to analyze the deployment costs of TAs and evaluate the amount of resources (e.g., computing, memory and bandwidth) that are required to handle a large number of TCP flows.

Acknowledgments. This work has been partially funded by MCIN/AEI/10.13039/501100011033 and by the European Union "Next GenerationEU /PRTR", by the Ministry of Science, Innovation and Universities (projects TED2021-130913B-I00, PDC2022-133465-I00), by the project PID2021-124054OB-C31 and the grant CAS21/00057 (MCI/AEI/FEDER, UE), and by the Regional Ministry of Economy, Science and Digital Agenda of the Regional Government of Extremadura (GR21133).

Declarations

Ethics Approval. This article does not contain any studies with human participants or animals performed by any of the authors.

Competing interests. The authors declare no competing interests, financial or otherwise.

Authors contributions. All authors contributed to the study conception and design. Material preparation, simulation and analysis were performed by all authors. All authors read and approved the final manuscript.

References

[1] Zhani, M.F., ElBakoury, H.: FlexNGIA: A flexible Internet architecture for the next-generation tactile Internet. Journal of Network and Systems Management, 1–45 (2020)

[2] Yahyaoui, H., Majdoub, M., Zhani, M.F., Aloqaily, M.: On Minimizing TCP Retransmission Delay in Softwarized Networks. In: IEEE/IFIP Network Operations and Management Symposium (NOMS), pp. 1–6 (2022). https://doi.org/10.1109/NOMS54207.2022.9789762

[3] Zhani, M.F., Boughamoura, R., Yahyaoui, H., Kaippallimalil, J., Kiani, A.: Oblivious TCP Support - A Virtual Network Function to Speed Up TCP in Wireless Environments. In: IFIP Wireless and Mobile Networking Conference (WMNC), pp. 75–79 (2022). https://doi.org/10.23919/WMNC56391.2022.9954287

[4] Transmission Control Protocol. RFC Editor (1981). https://doi.org/10.17487/RFC0793 . https://www.rfc-editor.org/info/rfc793

[5] Galán-Jiménez, J., Zhani, M.F., León, L.J.M., Kaippallimalil, J.: Transport Assistants to Enhance TCP Performance: Analysis of the Packet Delivery Delay. In: IEEE/IFIP Network Operations and Management Symposium (NOMS) (2024)

[6] Luglio, M., Roseti, C., Gerla, M.: The impact of efficient flow control and OS features on TCP performance over satellite links. ASSI Satellite Communication Letter (Sat-Comm Letter) **3**, 1–9 (2004)

[7] Wang, M.-H., Chen, L.-W., Chi, P.-W., Lei, C.-L.: SDUDP: A reliable UDP-Based transmission protocol over SDN. IEEE Access **5**, 5904–5916 (2017)

[8] Chen, J., Ye, Q., Quan, W., Yan, S., Do, P.T., Zhuang, W., Shen, X.S., Li, X., Rao, J.: SDATP: An SDN-based adaptive transmission protocol for time-critical services. IEEE Network **34**(3), 154–162 (2019)

[9] Chen, J., Yan, S., Ye, Q., Quan, W., Do, P.T., Zhuang, W., Shen, X.S., Li, X., Rao, J.: An SDN-based transmission protocol with in-path packet caching and retransmission. In: International Conference on Communications (ICC), pp. 1–6 (2019)

[10] Wan, C.-Y., Campbell, A.T., Krishnamurthy, L.: PSFQ: a reliable transport protocol for wireless sensor networks. In: ACM International Workshop on Wireless Sensor Networks and Applications, pp. 1–11 (2002)

[11] Spatscheck, O., Hansen, J.S., Hartman, J.H., Peterson, L.L.: Optimizing TCP forwarder performance. IEEE/ACM Transactions on Networking **8**(2), 146–157 (2000) https://doi.org/10.1109/90.842138

[12] Roşu, M.-C., Roşu, D.: An Evaluation of TCP Splice Benefits in Web Proxy Servers. In: ACM International Conference on World Wide Web (WWW), New York, NY, USA (2002)

[13] Mishra, T.K., Sahoo, K.S., Bilal, M., Shah, S.C., Mishra, M.K.: Adaptive congestion control mechanism to enhance tcp performance in cooperative iov. IEEE

Access **11**, 9000–9013 (2023) https://doi.org/10.1109/ACCESS.2023.3239302

[14] Pokhrel, S.R., Choi, J.: Improving tcp performance over wifi for internet of vehicles: A federated learning approach. IEEE transactions on vehicular technology **69**(6), 6798–6802 (2020)

[15] Bruhn, P., Kühlewind, M., Muehleisen, M.: Performance and improvements of tcp cubic in low-delay cellular networks. Computer Networks **224**, 109609 (2023) https://doi.org/10.1016/j.comnet.2023.109609

[16] León, L.J.M., Herrera, J.L., Berrocal, J., Galán-Jiménez, J.: Logistic Regression-based Solution to Predict the Transport Assistant Placement in SDN networks. In: IEEE/IFIP Network Operations and Management Symposium (NOMS 2023) (2023)

6 Tables

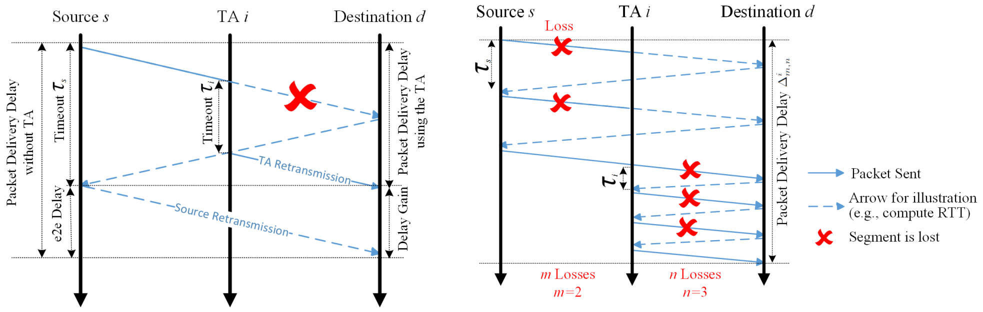
Table 3: Available paths from host h1 to host h2 in the Abilene Topology

Path ID	**Path**	**Number of nodes**	**Propagation delay (ms)**
$P1$	$(s1, s2, s5, s7, s4, s11)$	6	53.4
$P2$	$(s1, s2, s6, s7, s4, s11)$	6	67.8
$P3$	$(s1, s2, s5, s8, s10, s11)$	6	31.0
$P4$	$(s1, s2, s5, s7, s4, s10, s11)$	7	59.2
$P5$	$(s1, s2, s6, s7, s4, s10, s11)$	7	73.6
$P6$	$(s1, s2, s5, s8, s10, s4, s11)$	7	49.6
$P7$	$(s1, s2, s6, s7, s5, s8, s10, s11)$	8	78.6
$P8$	$(s1, s2, s12, s9, s3, s6, s7, s4, s11)$	9	82.0
$P9$	$(s1, s2, s6, s7, s5, s8, s10, s4, s11)$	9	97.5
$P10$	$(s1, s2, s12, s9, s3, s6, s7, s4, s10, s11)$	10	87.8
$P11$	$(s1, s2, s12, s9, s3, s6, s7, s5, s8, s10, s11)$	11	92.8
$P12$	$(s1, s2, s12, s9, s3, s6, s7, s5, s8, s10, s4, s11)$	12	111.4

Table 4: Description of the studied scenarios. Note that all scenarios consider the studied path P2 (highlighted in Fig. 13)

Scenario	Propagation Delay	Packet Loss
1	Delays of P2 shown in Fig. 13	10% for access links (h1,s1) and (s11,h2) 0% for other links
2	Delays of P2 shown in Fig. 13	Random loss between 0% and 10% for all links
3	Delays of P2 shown in Fig. 13	Random loss between 0% and 10% for all links
4	Same in all links ($5ms$)	Random loss between 0% and 10% for all links
5	Long delay close to the source	Same as scenario 4
6	Long delay close to the destination	Same as scenario 4

7 Figures



(a) Packet delivery delay with and without Transport Assistant

(b) Packet delivery assuming the segment is lost m times between the source s and the TA (node i) and lost n times between the TA and the destination d

Fig. 12: Packet delivery delay

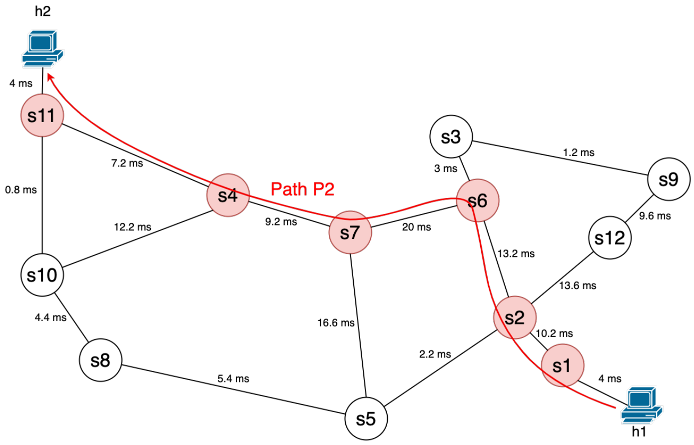


Fig. 13: Abilene topology. Links propagation delays follow a normal distribution with a mean of $8ms$ and standard deviation of $5ms$.

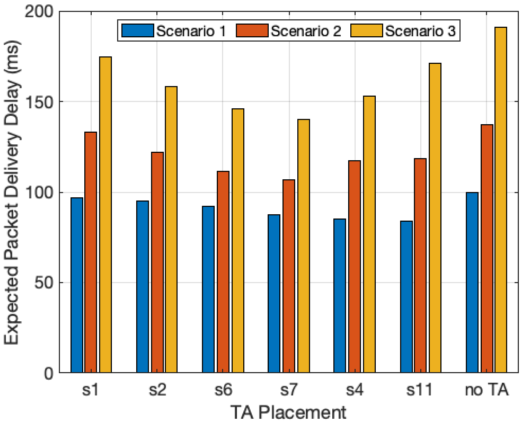


Fig. 14: Expected packet delivery delay for different TA placement in the path P2 for scenarios 1, 2 and 3 described in Table 4.

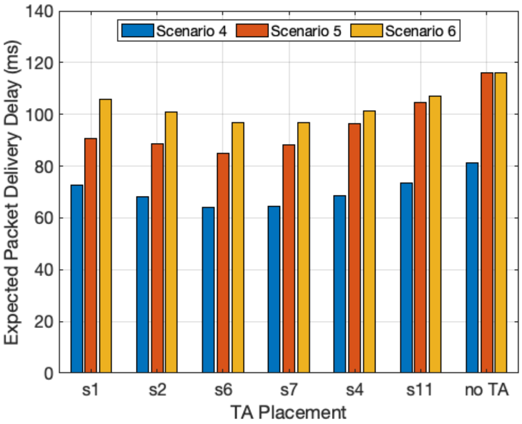


Fig. 15: Expected packet delivery delay for different TA placement in the path P2 for scenarios 4, 5, and 6 described in Table 4.

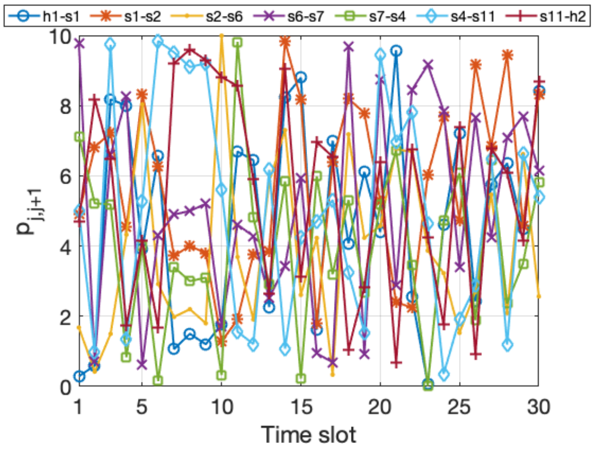


Fig. 16: Packet loss probability over time for all links composing the considered path P2.

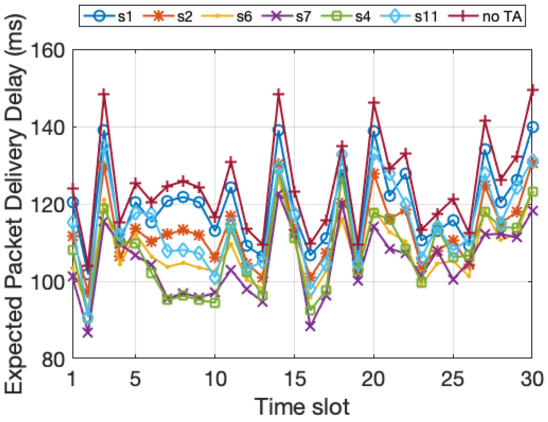


Fig. 17: Expected packet delivery delay over time for different TA placements.

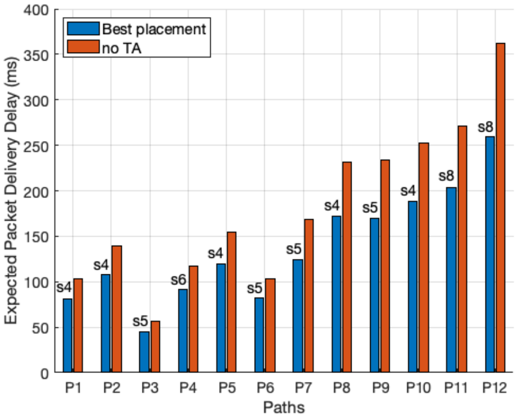


Fig. 18: Expected packet delivery delay for all paths between $h1$ and $h2$. Link propagation delays are the ones presented in Fig. 2.

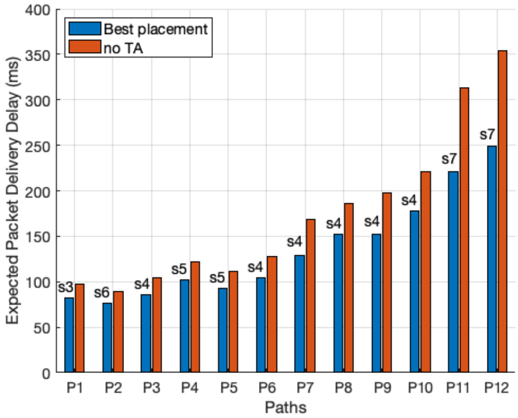


Fig. 19: Expected packet delivery delay for all paths between $h1$ and $h2$. Average link propagation delays equal to 8 ms.

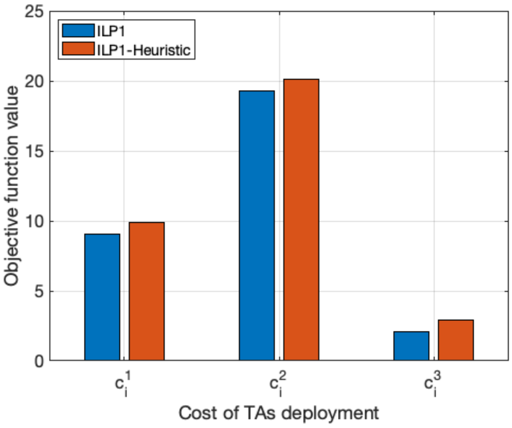


Fig. 20: Comparison between ILP1 and Heuristic 1 considering the three cost functions.

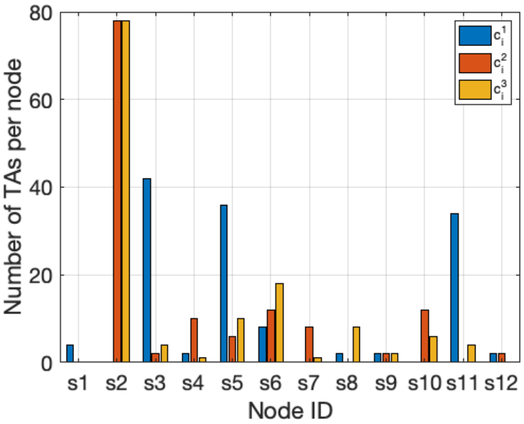


Fig. 21: Number of TAs per node for ILP1 considering the three cost functions.

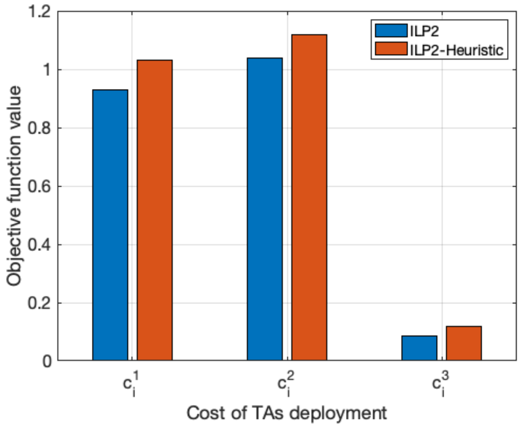


Fig. 22: Comparison between ILP2 and Heuristic 2 considering the three cost functions.