KARADENİZ TECHNICAL UNIVERSITY
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEPARTMENT OF COMPUTER ENGINEERING

MULTI TOPOLOGY ROUTING BASED TRAFFIC OPTIMIZATION FOR IEEE 802.1 TIME SENSITIVE NETWORKING

MASTER THESIS

Ömer Kağan DEMİR

JUNE, 2022
TRABZON

KARADENİZ TECHNICAL UNIVERSITY

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

DEPARTMENT OF COMPUTER ENGINEERING

MULTI TOPOLOGY ROUTING BASED TRAFFIC OPTIMIZATION FOR IEEE 802.1 TIME SENSITIVE NETWORKING

Ömer Kağan DEMİR

This thesis is accepted to give the degree of

"MASTER OF SCIENCE"

By

The Graduate School of Natural and Applied Sciences at
Karadeniz Technical University

The Date of Submission : **24 / 05 /2022**

The Date of Examination : **24 / 06 /2022**

Supervisor : **Asst. Prof. Selçuk CEVHER**

Trabzon 2022

ACKNOWLEDGEMENT

In this study, a novel algorithm to compute *Audio-Video Bridging* (AVB) routes in a *Time Sensitive Network* (TSN) network is proposed using the *Multi Topology Routing* (MTR) concept. A variety of traffic scenarios are created using real network topologies to perform the experiments. I would like to thank my advisor Asst. Prof. Selçuk CEVHER for his interest, assistance and guidance. Finally, i would like to thank my family for their financial and moral support throughout my work.

Ömer Kağan DEMİR

Trabzon 2022

STATEMENT OF ETHICS

The research work in this thesis entitled “MULTI TOPOLOGY ROUTING BASED TRAFFIC OPTIMIZATION FOR IEEE 802.1 TIME SENSITIVE NETWORKING” is a report of reliable research carried out by me, Ömer Kağan DEMİR, and under supervision and directions of Asst. Prof. Dr. Selçuk CEVHER. I accept that the experiments represented in this work were conducted by me, the information’s taken from the other research were referenced in the text, the scientific and ethical rules were not violated. 24/06/2022.

Ömer Kağan DEMİR

CONTENTS

Yüksek Lisans Tezi

ÖZET

IEEE 802.1 ZAMAN HASSAS HABERLEŞME İÇİN ÇOKLU TOPOLOJİ YÖNLENDİRMESİ TABANLI TRAFİK OPTİMİZASYONU

Ömer Kağan DEMİR

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Dr. Öğr. Üyesi Selçuk CEVHER
2022, 59 Sayfa

Endüstriyel otomasyon, araç-içi ve aviyonik haberleşme platformları gibi Siber Fiziksel Sistemlerdeki fiziksel süreçlerin etkin yönetimi için deterministik gerçek-zamanlı bir haberleşmeye ihtiyaç duyulmaktadır. IEEE 802.1 *Zaman Hassas Haberleşme* (ZHH) çalışma grubu, Ethernet tabanlı deterministik haberleşme teknolojilerinin standartlaştırılmasını hedefleyen başlıca organizasyondur. IEEE 802.1 ZHH, *Ses-Video Köprüleme* (SVK) ve *Zaman Tetiklemeli* (ZT) trafik sınıflarını desteklemektedir. Bu çalışmada, ZT ve SVK akışlarının birlikte iletildiği gerçek bir ağ topolojisi kullanılarak, arama uzayı azaltımı için *Çoklu Topoloji Yönlendirmesi* (ÇTY) tabanlı bir trafik optimizasyonu geliştirilmiş ve *Aç Gözlü Rastgele Ayarlanabilir Arama Prosedürü* (AGRAAP) meta-sezgisel yaklaşımı kullanılarak SVK akışlarının gerçek-zaman gereksinimlerinin karşılanması hedeflenmiştir. ÇTY yaklaşımı sayesinde, ZT güzergahları tarafından içerilen linklerin yüksek link ağırlıkları kullanılarak izole edildiği, fiziksel topolojinin bir kopyası (sanal topoloji) oluşturulmuştur. Performans sonuçları, önerilen yöntemin geleneksel yaklaşıma kıyasla çizelgelenemeyen SVK akış sayısını maksimum %60 oranında azalttığını göstermektedir.

Anahtar Kelimeler: Çoklu Topoloji Yönlendirmesi, Siber Fiziksel Sistem, Deterministik Haberleşme, IEEE 802.1 ZHH Standardı, Güzergah Planlama

Master Thesis

SUMMARY

MULTI TOPOLOGY ROUTING BASED TRAFFIC OPTIMIZATION FOR IEEE 802.1 TIME SENSITIVE NETWORKING

Ömer Kağan DEMİR

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Computer Engineering Graduate Program
Supervisor: Asst. Prof. Selçuk CEVHER
2022, 59 Pages

A deterministic real-time communication is required by the effective management of physical processes in *Cyber Physical Systems* (CPS) including industrial automation, in-vehicle and avionic communication platforms. IEEE 802.1 *Time Sensitive Network* (TSN) task group is the leading organization that aims to standardize Ethernet-based deterministic communication technologies. IEEE 802.1 TSN supports *Audio-Video Bridging* (AVB) and *Time-Triggered* (TT) traffic classes. In this study, a *Multi-Topology Routing* (MTR) based traffic optimization is developed for search space reduction using a real network topology where TT and AVB streams are simultaneously transmitted. Furthermore, it is aimed to satisfy the real-time requirements of AVB streams by utilizing *Greedy Randomized Adaptive Search Procedure* (GRASP) meta-heuristic approach to perform local search. Thanks to the MTR approach, a copy of the physical topology (virtual topology) is created, in which the links traversed by the TT routes are isolated using very high link weights. The performance results show that the proposed approach reduces the number of unschedulable AVB streams by up to 60% compared to the conventional approaches.

Key Words: Multi Topology Routing, Cyber-Physical Systems, Deterministic Communications, IEEE 802.1 TSN, Route Planning

LIST OF FIGURES

Page No

LIST OF TABLES

Page No

LIST OF SYMBOLS

E	Set of Edges
ES	Set of End Systems
$cost(\mathcal{R}_{local})$	Cost of $\mathcal{R}_{local}$
$cost(\mathcal{R}_{best-so-far})$	Cost of $\mathcal{R}_{(\mathcal{R}_{best-so-far})}$
$W(R_A)$	Total weight of R_A
CR_{all}	Candidate Routes of all AVB streams
$W(R_S)$	Total weight of R_S
CR_s	Candidate Routes of s
$\boldsymbol{R}$	Route
R_A	Alternative route of s flow
$R_{[a,b]}$	Link rate of edge transmits packet from b to a
R_{PHY}	Routes returned by the KSP algorithm from the physical topology
R_S	Route of s flow
R_{VT}	Routes returned by the KSP algorithm from the virtual topology
$\boldsymbol{P}$	Potential Route
S_{AVB}	Set of AVB flows transmitted in the network
S_{TT}	Set of TT flows transmitted in the network
SW	Set of Switches
TO_{MTR}	Our MTR based AVB traffic optimization approach
TO_{PHY}	AVB traffic optimization approach that considers only the physical topology
V	Set of Vertexes
$w(l)_{VT}$	Weight of link l of TT stream in virtual topology
$\omega_{[a,b]}$	Propagation delay of edge transmits packet from b to a
$\mathcal{R}$	Routing Solution
w_{high}	The weight of a TT link
$\|E\|$	Total number of edges
w_{max}	Maximum of the normal weight values in the physical topology
$\mathcal{R}_{(\mathcal{R}_{best-so-far})}$	Best routing solution

$\mathcal{R}_{ini}$	Initial routing solution
$\mathcal{R}_{local}$	The routing solution returned by GRASP

LIST OF ABBREVIATIONS

AVB	Audio-Video Bridging
AGRAAP	Aç Gözlü Rastgele Ayarlanabilir Arama Prosedürü
BE	Best Effort
CBS	Credit Based Shaper
CNC	Centralized Network Configuration
CPS	Cyber Physical System
CUC	Centralized User Configuration
ÇTY	Çoklu-Topoloji Yönlendirme
FKSPL	Finalized K-Shortest Paths List
GCL	Gate Control List
gPTP	Generalized Precision Time Protocol
GRASP	Greedy Randomized Adaptive Search Procedure
ILP	Integer Linear Programming
IS-IS	Intermediate System - Intermediate System
K-EKG	K-En Kısa Güzergah
KSP	K-Shortest Path
MAC	Medium Access Control
MTR	Multi Topology Routing
MTU	Maximum Transmission Unit
OSPF	Open Shortest Path First
PKSPL	Potential K-Shortest Paths List
PTP	Precision Time Protocol
SPF	Shortest Path First
SVK	Ses-Video Köprüleme
TSN	Time Sensitive Networking
TT	Time-Triggered
UBS	Urgency Based Scheduler
UNI	User/Network Configuration Info

VT	Virtual Topology
WCD	Worst Case Delay
ZHH	Zaman Hassas Haberleşme
ZT	Zaman Tetiklemeli

1. INTRODUCTION

The physical processes in *Cyber Physical Systems* (CPS) such as industrial automation, in-vehicle and avionic communication platforms are managed by control computers via sensors and actuators. In order to deliver the sensor data to control computer and the control messages to actuators in a timely fashion, a deterministic real-time communication guaranteeing a bounded end-to-end latency and a low jitter is needed [1]. The recent research work has focused on the real-time variants of IEEE 802.3 Ethernet such as TTEthernet [2], SERCOS III [3] and Profinet [4] to be able to satisfy the increasing bandwidth and low-cost requirements of CPS communication platforms [5]. The inter-operability issue of the current real-time Ethernet variants necessitates the standardization of Ethernet-based deterministic real-time technologies.

IEEE 802.1 TSN task group is the leading organization which targets the standardization of Ethernet-based deterministic real-time communication technologies. IEEE 802.1 *Time Sensitive Networking* TSN enables the transmission of traffic classes with varying time criticalities, namely *Time-Triggered* (TT), *Audio-Video Bridging* (AVB) and *Best-Effort* (BE), over the same network infrastructure [5]. IEEE 802.1Qbv standard specifies a mechanism aiming at prioritizing the forwarding of TT traffic on network switches, which has the most stringent real-time requirements. This mechanism relies on the utilization of scheduling tables called *Gate Control Lists* (GCL) each of which determines the time offsets for an outgoing port at which TT frames are placed onto the transmission line. AVB streams, which have less stringent timing requirement, may be categorized into two different classes, namely *Class A* or *Class B*, with different priorities. An AVB frame is allowed to be transmitted on an outgoing port only if there exists no frames with a higher priority awaiting to be transmitted through the same port. Time-critical CPS communication platforms can be statically configured at design time since the properties of the traffic flows are known in advance of the actual network operation [6].

The transmission routes of real-time streams should be intelligently determined in order to satisfy the timing requirements in a TSN network [7]. Efficiently balancing the real-time traffic load across the network links resolves the congestions taking place at network switches, and, hence, reduces the interference of the streams transmitted through the same outgoing port to each other. In this study, a *Multi-Topology Routing* (MTR) based traffic

optimization is developed for search space reduction using a real network topology where TT and AVB streams are simultaneously transmitted. Furthermore, it is aimed to satisfy the real-time requirements of AVB streams by utilizing *Greedy Randomized Adaptive Search Procedure* (GRASP) meta-heuristic approach to perform local search. Thanks to the MTR approach, a copy of the physical topology (virtual topology) is created, in which the links traversed by the TT routes are isolated using very high link weights. The proposed approach computes *K/2* shortest paths with an increasing length between the endpoints of each AVB stream using the physical topology while remaining *K/2* shortest paths are generated using the virtual topology. GRASP chooses the best route from these candidate paths minimizing an intelligently defined cost function. The reason why selecting from a set of candidate routes with an increasing length instead of solely utilizing the shortest route is because shortest-path routing causes bottleneck network links to arise with a high probability [6, 8]. In order to determine the worst-case transmission delays *Worst-Case Delay* (WCD) caused by the proposed approach, the *AVB Latency Math* theoretical analysis tool defined in the IEEE 802.1BA [9] standard is used. This algorithm determines WCD for an AVB stream by identifying the worst-case scenario that will cause the highest latency for the stream. The worst-case scenario for each stream is determined by considering other streams (TT and AVB) that share the stream's transmission route. It should be noted that the WCD will increase as the number of flows sharing a transmission route increases. The WCD amount of a flow is computed by summing the WCD values of each network component on the selected route.

1.1. IEEE 802.1 Time Sensitive Networking

Table 1.1. IEEE 802.1 TSN Protocols

Synchronization	Latency
• Timing and Synchronization [802.1AS-2020] • Hot Standby [P802.1ASdm] • YANG [P802.1ASdn] • Inclusive Terminology [P802.1ASdr]	• Credit Based Shaper [802.1Qav] • Frame Preemption [802.1Qbu & 802.3br] • Scheduled Traffic [802.1Qbv] • Cyclic Queuing and Forwarding [802.1Qch] • Asynchronous Traffic Shaping [802.1Qcr] • Shaper Parameter Settings [P802.1Qdq] • QoS Provisions [P802.1DC]
Resource Management	**Reliability**
• Stream Reservation Protocol [802.1Qat] • Link-local Registration Protocol [802.1CS] • TSN Configuration [802.1Qcc] • YANG for CFM [802.1Qcx] • YANG for LLDP [P802.1ABcu] • YANG for 802.1Qbv/Qbu/Qci [P802.1Qcw] • YANG & MIB for FRER [P802.1CBcv] • Extended Stream Identification [P802.1CBdb] • Resource Allocation Protocol [P802.1Qdd] • TSN Configuration Enhancements [P802.1Qdj] • LLDPv2 for Multiframe Data Units [P802.1ABdh] • Multicast and Local Address Assignment [P802.1CQ]	• Frame Replication and Elimination [802.1CB] • Path Control and Reservation [802.1Qca] • Per-Stream Filtering and Policing [802.1Qci] • Reliability for Time Sync [802.1AS-2020]

Table 1.1, based on [10], shows the TSN protocols. Protocols can be examined under four main classes according to their functionalities. The set of protocols that have P before their name are not standardized yet. The fundamental protocols for a TSN network are explained in the following sections:

1.1.1. IEEE 802.1Qbv - Scheduled Traffic

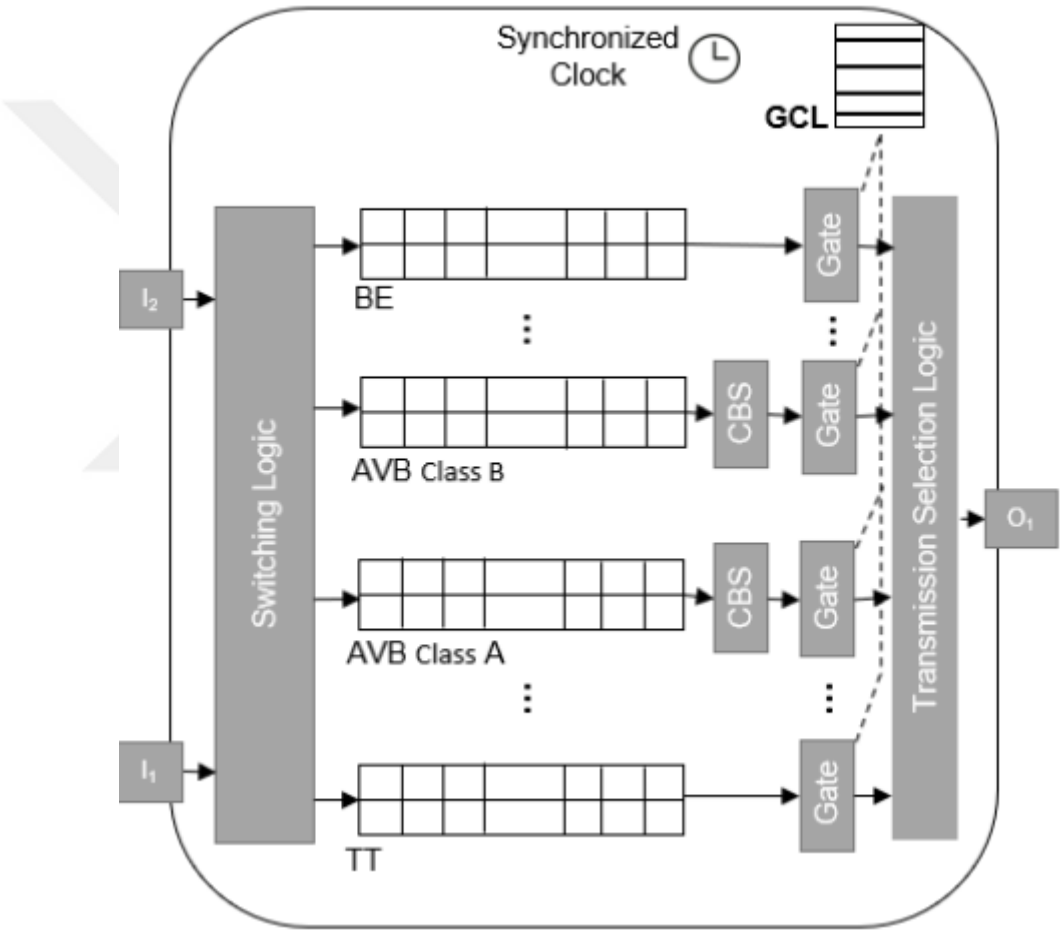


Figure 1.1. Architecture of a TSN enabled switch

Figure 1.1. shows the structure of an 802.1Qbv compatible TSN switch with two input ports (I_1, I_2) and one output port (O_1). The 802.1Qbv defines eight priority queues for each output port and defines a configurable on/off gate at the output of each queue. The timing of the state changes of the gates on the output ports is stored in scheduling tables called GCL synchronized with the 802.1ASrev time synchronization standard. Priority queues are allocated offline starting from the highest-priority traffic class, namely, TT, AVB Class A/B and BE [8]. *Switching Logic* decides the relevant output port for traffic flows and forwards the frames to the corresponding priority queue, considering their priority status. *Transmission Select Logic* allows transmission of a frame waiting in the priority queue only if the corresponding gate is open.

1.1.2. IEEE 802.1Qav - Credit Based Shaper

AVB is a traffic type defined in the IEEE 802.1Qav standard for use in soft real time communication. For the transmission of AVB flows, the *Credit Based Shaper* (CBS) mechanism defined in the IEEE 802.1Qat standard, located between the relevant priority queue and the port, is used. For an AVB frame to be transmitted, the gate of the corresponding priority queue must be open, there must be no higher priority frames waiting in other queues, and the CBS mechanism must allow transmission. The purpose of the CBS mechanism is to limit the transmission frequency of AVB frames in the event of a burst in AVB traffic, thus allowing lower priority traffic types to be transmitted [8].

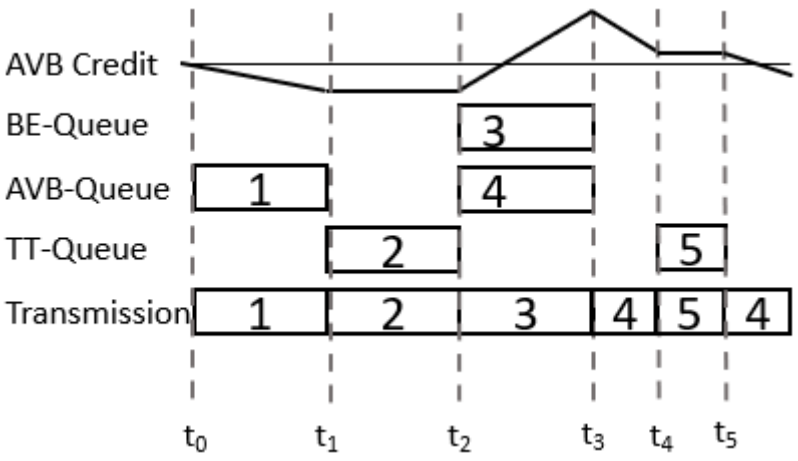


Figure 1.2. Example of BE, AVB and TT transmissions

Figure 1.2. inspired by [11], shows the operation of CBS in a scenario where TT, AVB and BE streams are simultaneously transmitted. At t_0, AVB $frame_1$ arrives at the AVB queue. Since the AVB frame can be transmitted when the credit is zero or positive, AVB $frame_1$ is placed on the transmission line. At t_1, the transmission of the AVB $frame_1$ is completed and the credit is now negative. At t_1, TT $frame_2$ arrives in the TT queue and is placed on the transmission line. Credit is frozen because the AVB queue is closed. When the transmission of the TT $frame_2$ is finished at t_2, BE $frame_3$ and AVB $frame_4$ arrive in the corresponding queues. Since the credit is negative, AVB $frame_4$ cannot be placed on the transmission line. BE $frame_3$ is placed on the transmission line and credits are started to be increased. At t_3, since the transmission of the BE $frame_3$ is finished and the credit is positive, the AVB $frame_4$ is put on the transmission line. Credit begins to decrease. During the transmission of AVB $frame_4$, TT $frame_5$ comes to the relevant queue at the time of t_4 and the transmission of AVB $frame_4$ is interrupted, TT $frame_5$ is placed on the transmission line. Credit is frozen at this time. After the transmission of the TT $frame_5$ is finished at the time of t_5, the transmission of the AVB $frame_4$ and credit decrease continues.

1.1.3. IEEE 802.1AS/ASrev - Timing and Synchronization

This standard is a collection of procedures created to meet time-synchronization requirements for time-sensitive applications [12]. *The Precision Time Protocol* (PTP), defined in IEEE 1588-2008, is a standard used to provide clock synchronization at the physical layer. IEEE 802.1AS is a protocol based on IEEE 1588. IEEE 802.1AS or *Generalized Precision Time Protocol* (gPTP) has some features for AVB applications unlike PTP [13, 14].

Clocks on switches can be out of sync due to various reasons. The Grandmaster defined in IEEE 802.1AS keeps the reference time. The grandmaster provides synchronization by transmitting the time information to the slave nodes in a periodic time interval. Time synchronization is essential in a TSN network in order to synthesize GCLs. In the case of the slightest synchronization disorder, conflicts will begin in the TT traffic transmission and deterministic communication will be disrupted.

1.1.4. IEEE 802.1Qbu & 802.3br - Frame Preemption

Frame preemption has an important role in deterministic communication. While a low priority frame is being transmitted, when a high priority frame enters the transmission line, the transmission of the low priority packet should be stopped and the transmission of the low priority packet should be resumed after the transmission of the high priority packet is finished. In Figure 1.2. it is seen that TT $frame_5$ preempts AVB $frame_4$ between t_4 and t_5.

Therefore, IEEE 802.1Qbu, which has become a standard with IEEE 802.1Q-2018, provides the following features for this TSN:

- Interrupting or canceling the transmission of the transmitted frame to transmit higher priority frames from the transmitted frame.
- Resuming the transmission of the interrupted frame after the transmission of the high priority frame is complete [15].

IEEE 802.3br includes procedures for separating smaller pieces of non-time critical frames being transmitted while time-critical frames are transmitted on the physical link. For this purpose, IEEE 802.3br defines optional *Medium Access Control* (MAC) sublayer. This

sublayer and the MACs attached to the sublayer allow to realize the features described above [15].

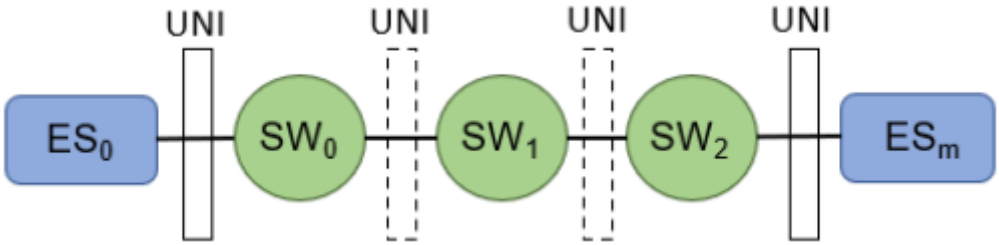
1.1.5. IEEE 802.1CB - Frame Replication and Elimination

Faults may occur in the route of the packet from a talker to a listener. This packet must be delivered to its destination by a different route so that communication is not interrupted. For this reason, IEEE 802.1CB, which has become a standard with IEEE 802.1Q-2018, has been created. IEEE 802.1CB defines the sequence number for each frame in a stream. Each frame is replicated according to this sequence number. The stream is split into small parts and sent to the listener in different ways. The splitted stream is combined in different parts of the network, and duplicate packets are destroyed using the sequence number [16]. Due to this mechanism, it provides static redundancy [17].

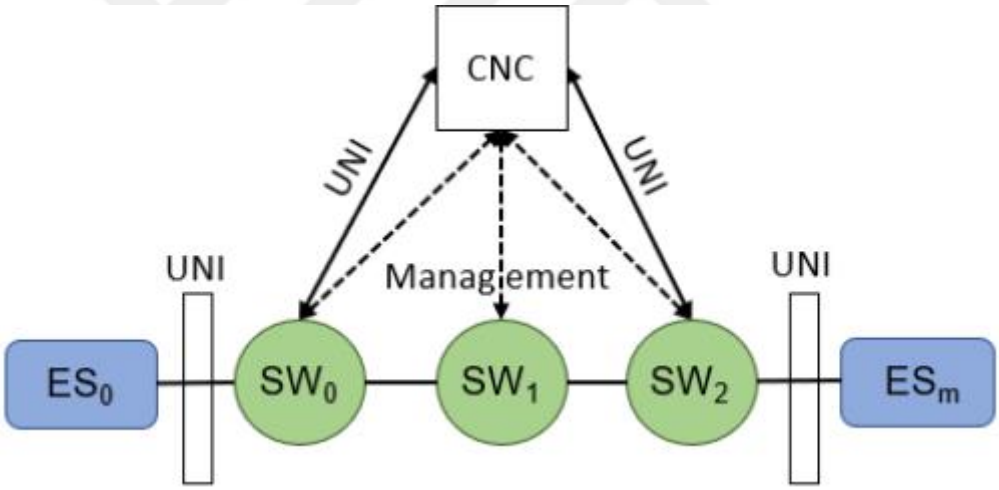
1.1.6. IEEE 802.1Qcc - TSN Configuration

Figure 1.3., inspired by [18], shows the IEEE 802.1 TSN configuration types. These configurations have pros and cons. Figure 1.3a shows the *fully distributed model*. In this model, the *User/Network Configuration Info* (UNI) is located between end systems and switches. End System sends the properties of the stream to the switch it is connected to via UNI. UNI is propagated by the switches to the end system where the stream will arrive (dashed rectangle). When UNI reaches the listener, it transmits the information about whether the stream is ready to be received or not, to the talker in the same way. If the listener is ready to receive the stream, the switches upgrade their configuration according to the properties of the stream [15]. Figure 1.3b shows the *centralized* model. UNI continues to exist between the end systems and the switch that connected to end systems. Unlike fully distributed, switches are not configured locally. *Centralized Network Configuration* CNC uses UNI to configure the switch using the remote network management protocol. In Figure 1.3b, end systems send the information of the stream to the switch with UNI (rectangle). These switches send properties of the stream to the CNC (solid arrow). CNC configures the switches using the remote network management protocol (dashed arrow) [18]. Figure 1.3c

shows the *fully centralized model*. In this model, there is no UNI between the end systems and the switches. *Centralized User Configuration* (CUC) discovers and learns the features of end systems. CUC sends configuration information to the CUC via UNI to configure the switches [18]. Our traffic optimization method can be integrated to CNC and calculated routes can be deployed to switches via CNC.



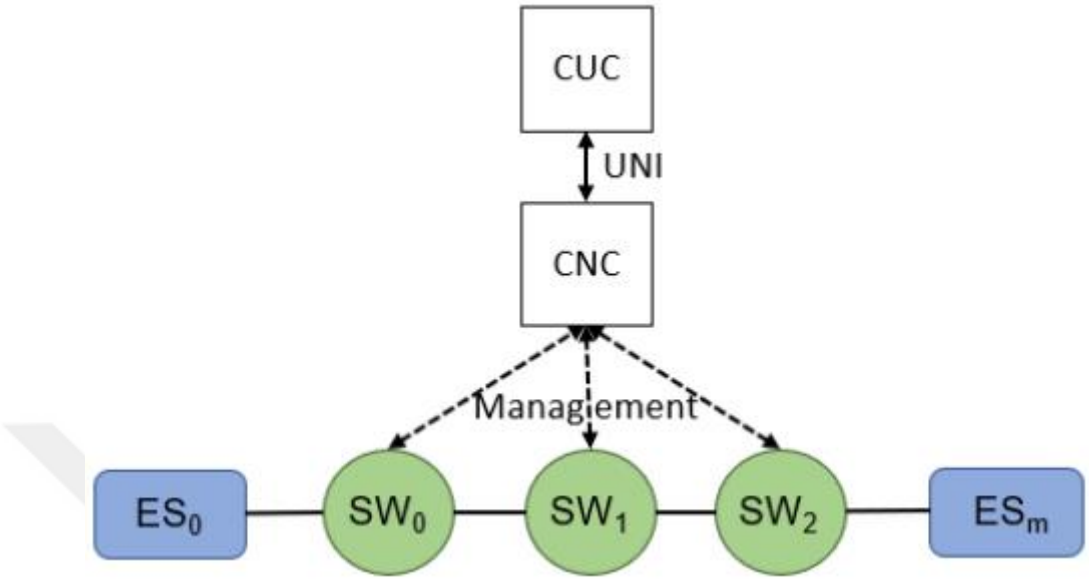
(a) Fully distributed model



(b) Centralized model

Figure 1.3. TSN configuration models

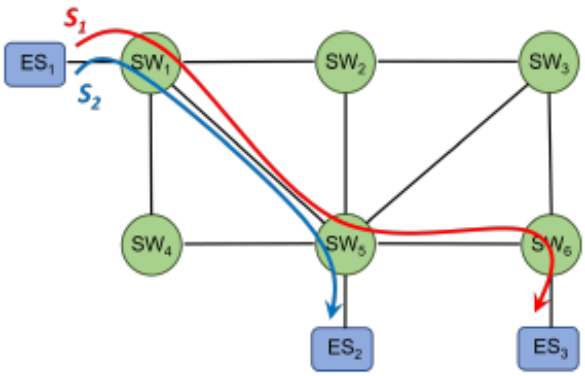
continuation of Figure 1.3.



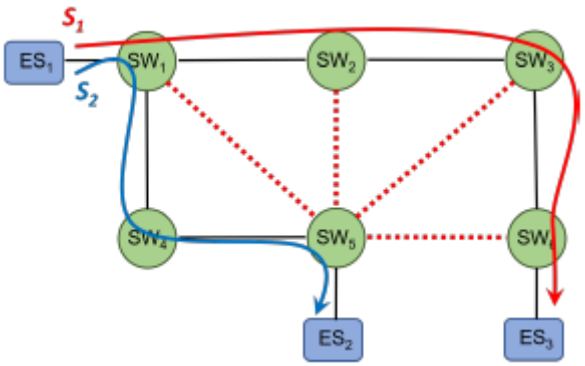
(c) Fully centralized model

1.2. Multi Topology Routing

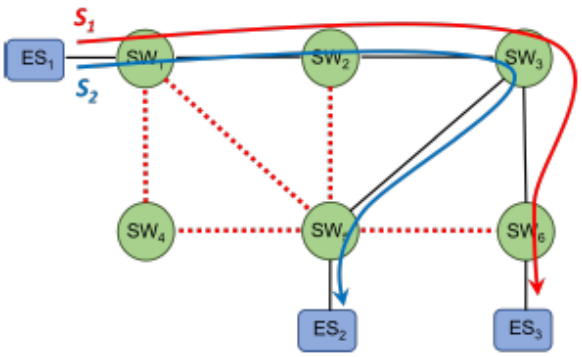
Internet Engineering Task Force standardized the MTR concept as extensions to *Open Shortest Path First* (OSPF) and *Intermediate System - Intermediate System* (IS-IS), which allows the configuration of service differentiation through class-based forwarding [19, 20]. MTR supports *virtual topologies* (VT) which have the same network graph as the physical topology, but with different link weights. Since each VT provides a separate forwarding capability due to its specific link weight setting, MTR leads to a significant amount of flexibility in making policy-based routing decisions by defining separate VTs for voice, video, and data traffic classes to follow specific transmission paths. Thanks to this flexible design of alternate VTs, the policy-based routing capability of the MTR concept enables achieving a desired level of traffic load balancing even in the face of link failures [21, 22].



(a) Physical Topology



(b) VT 1



(c) VT 2

Figure 1.4. Example VTs computed for a physical topology

1.3. K-Shortest Paths

In our thesis, Yen algorithm is used to compute *K-Shortest Paths* (KSP). The Yen algorithm calculates KSP while ensuring that paths are loopless [23, 24].

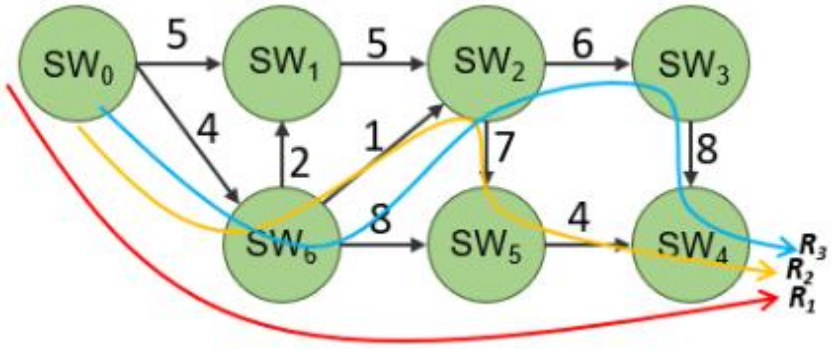


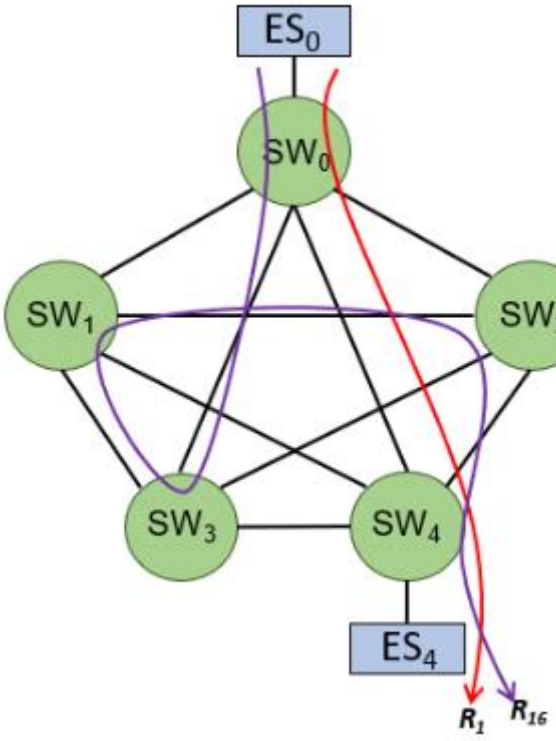
Figure 1.5. Routes that the Yen algorithm computes

Figure 1.5., inspired by [25], shows the shortest paths from SW_0 to SW_4 computed using the Yen algorithm (K = 3). The flow of the Yen algorithm is as follows:

1. The first path (shortest) is calculated using the Dijkstra algorithm *Route₁* (***R₁***). ***R₁*** is added to the *Finalized K-Shortest Paths List* (FKSPL). The links of the paths in this list are set to infinite weight, respectively, and Dijkstra is run again. In this case ***R₁*** computed as $SW_0 \rightarrow SW_6 \rightarrow SW_5 \rightarrow SW_4$ and cost equals to 16.
2. $SW_0 \rightarrow SW_6$ is set to infinite weight, *Potential Path₁* (***P₁***) computed as $SW_0 \rightarrow SW_1 \rightarrow SW_2 \rightarrow SW_5 \rightarrow SW_4$ and cost equals to 21. ***P₁*** is added to *Potential K-Shortest Paths List* (PKSPL).
3. $SW_6 \rightarrow SW_5$ is set to infinite weight, ***P₂*** computed as $SW_0 \rightarrow SW_6 \rightarrow SW_2 \rightarrow SW_5 \rightarrow SW_4$ and cost equals to 16. ***P₂*** is added to *PKSPL*.
4. $SW_5 \rightarrow SW_4$ is set to infinite weight, ***P₃*** computed as $SW_0 \rightarrow SW_6 \rightarrow SW_2 \rightarrow SW_3 \rightarrow SW_4$ and cost equals to 19. ***P₃*** is added to *PKSPL*. When we look at the *PKSPL*, the path with the lowest cost is determined as the second shortest path (***P₂***) and added to the *FKSPL*. ***P₂*** is now became ***R₂***.
5. After adding ***R₂*** to the *FKSPL*, the same operations are performed with the links above (***P₁***, ***P₂***, ***P₃*** are recomputed). With the addition of ***R₂*** to the *FKSPL*,

two new links should also be evaluated. $SW_6 \rightarrow SW_2$ is set to infinite weight, $\boldsymbol{P_4}$ computed as $SW_0 \rightarrow SW_6 \rightarrow SW_5 \rightarrow SW_4$ and cost equals to 16 (already in the *FKSPL*).

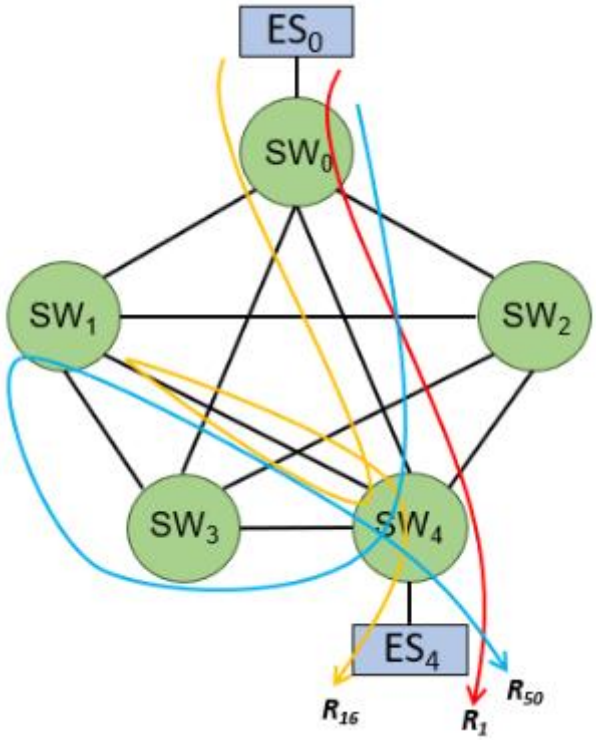
6. $SW_2 \rightarrow SW_5$ is set to infinite weight, $\boldsymbol{P_5}$ computed as $SW_0 \rightarrow SW_6 \rightarrow SW_5 \rightarrow SW_4$ and cost equals to 16 (already in the *FKSPL*). The path that is not in the *FKSPL* and has the lowest cost ($\boldsymbol{P_3}$) is determined as the third shortest path. $\boldsymbol{P_3}$ is now became $\boldsymbol{R_3}$.



(a) K-Shortest Paths computed by Yen algorithm

Figure 1.6. Example of computed K-Shortest Paths by Yen and Eppstein algorithms

continuation of Figure 1.6.



(b) K-Shortest Paths computed by Eppstein algorithm

There are multiple KSP algorithms. One of them is the Eppstein KSP algorithm [26]. Figure 1.6. shows the K-Shortest Paths computed by Yen and Eppstein in a complete graph. Links are assumed to be bidirectional. To show the difference of paths computed by Yen and Eppstein, assume that fifty shortest paths will be computed in the topology in Figure 1.6. Since the paths computed by Yen are loopless, Yen will not be able to compute fifty paths. The first and last path computed by Yen is shown in Figure 1.6a. Since the paths computed by Eppstein contain loops [26, 27] Eppstein computed fifty paths. Figure 1.6b shows the paths computed by Eppstein. When we look at $\boldsymbol{R_{16}}$, it is seen that it contains a loop, different from the path the Yen algorithm computes.

1.4. Contributions

The current literature on AVB route planning in a TSN network mostly uses the KSP algorithm to compute candidate routes from which the ultimate routes for AVB streams will be selected. The KSP algorithm computes the shortest paths with an increasing length between the source and destination of an AVB stream, starting from the shortest one. Using the candidate routes determined with the KSP algorithm, the solution minimizing the cost function is determined by utilizing an optimization approach. The KSP algorithm as used in the current literature operates according to link weights in physical topology and does not consider the TT streams passing through the links. In a network topology where TT and AVB streams are transmitted together, especially in a topology with a large number of TT streams, the candidate routes computed by KSP to be considered by the local search procedure are likely to heavily contain TT links. Our approach reduces the search space for GRASP to consider only the KSP of each AVB stream, half of which is computed using the physical topology while the other half is determined using a VT. Our approach constructs this VT by assigning a very high weight to each link in the VT in order to isolate it from the AVB routes if the link is used to transmit a TT stream so that the interference from the TT streams is attempted to be limited. In this way, the disadvantage of the conventional KSP algorithm taking into consideration solely the physical topology is attempted to be eliminated.

2. LITERATURE REVIEW

The work in [6] determines routes that will minimize the worst-case delay for AVB streams while using the shortest routes for TT. Considering the possible interference of TT streams with AVB streams, an approach that determines transmission routes and scheduling tables providing time constraints for TT streams is proposed in the study in [7]. In the study, the search space is reduced with a KSP based heuristic algorithm for route planning and optimum routes are tried to be determined by using the meta-heuristic approach of GRASP. The study in [8] uses a similar approach for route planning of AVB flows. In the study, the AVB Latency Math theoretical analysis tool is improved to consider the interference of TT streams with AVB streams. The study in [11] route computation and schedulability analysis are performed by considering only TT flows. Instead of the shortest path algorithm, two different heuristic algorithms are proposed and results are obtained in more than one topology using the GRASP meta-heuristic approach. The study [28] the network topology supporting the seamless transmission of TT streams was synthesized. While synthesizing this network topology, routing and schedulability synthesis of TT flows were also made. While creating the network topology, the disjoint route of the TT flows, which provides the feasible scheduling, is computed and the network cost is minimized. Yen KSP algorithm was used for finding disjoint routes. The study in [29] the network topology supporting the seamless transmission of streams with *Urgency-Based Scheduler* (UBS) traffic type was synthesized. Heuristic, GRASP and constraint programming methods are proposed and performance comparisons are made to determine the routing that meets the requirements of the UBS stream. Considering the time in the study, it was stated that GRASP is giving the best solution in a reasonable time. The study in [30] a heuristic-based reliability-aware multipath routing algorithm is proposed to increase the schedulability of TT flows in a TSN network. The heuristic algorithm that determines the candidate routes that meet the reliability constraint defined in the article has been developed, and the *Ant Lion Optimizer* meta-heuristic algorithm is used to find the result that minimizes the cost function. The algorithm is tested on real test topologies. The study in [31] it was desired to see the effect of the routes of the TT flows in the IEEE 802.1Qbv network on the schedulability. For this purpose, an *Integer Linear Programming* (ILP) based route algorithm has been developed by determining the parameters of the route algorithm that affect the schedulability. Thanks to

the ILP based algorithm, the slack in the schedule is used more efficiently, thus allowing more TT flows to be transmitted in the same network. The study in [32] proposes an integrated approach based on the use of ILP for route planning and scheduling computation of TT flows. An ILP approach is also coded for route planning that considers maximum link usage and route lengths together. It was concluded that as the maximum link utilization value increases, the schedulability decreases, and the transmission delays may increase if the route lengths increase. The study [33] proposes two heuristics for load balancing in TSN. The first of these approaches is trying to determine the route planning offline, which minimizes an objective function that includes the traffic load and route lengths on the links used by the flow routes; the other approach aims to eliminate bottleneck links by detecting the bottleneck links dynamically (online) that will occur if the routes are determined using Shortest Path First (SPF) and by moving the flows passing through the bottleneck links to suitable alternative routes. The study in [21], it is stated that if the OSPF routes of the flows are calculated using the standard shortest path algorithm, it may not be possible to distribute the traffic load evenly to the links. To eliminate this problem, they recommend using MTR which provides more flexibility in terms of load balancing. A heuristic approach is suggested in the study; in the proposed approach, the traffic flows to be transmitted are handled in order. For the first flow, the shortest path is computed with the Dijkstra algorithm, and the costs of the links are updated by determining the residual capacities of the links on the path. A testbed containing Cisco routers was used as the experimental environment. The rate of traffic successfully received by the hosts was used as a performance metric in the experiments, and MTR and standard OSPF were compared. MTR has been shown to perform better. The study in [34], a MTR approach that performs load balancing by taking into account the amount of traffic that changes according to the days suggested for a GEANT network. The Multi Topology part of this algorithm was created on the algorithm called Resilient Routing Layer. Two heuristic algorithms based on this approach have been proposed. Global Load Balancing, one of the proposed algorithms, can be used for load balancing in a real-time communication network.

3. SYSTEM MODEL

3.1. Application Model

In this thesis, applications generating periodic AVB and TT streams are considered. Each stream transmitted in the network is characterized by a 4-tuple of (*D*, *S*, *T*, *R*) where *D*, *S*, *T* and *R* denote the maximum allowed end-to-end delay, message size, message period and transmission route originating at a source ES and ending at one or more destination ESs, respectively. If *S* is larger than the *Maximum Transmission Unit* (MTU), namely 1500 bytes, the message is fragmented into multiple frames. Each TT stream is additionally associated with a set of GCLs each of which assigns a time offset to each frame of the stream for each link on its transmission route, which specifies when the respective frame will be transmitted on the corresponding outgoing port. The best route for each AVB stream is computed by our optimization approach while the routes and GCLs for the TT streams are assumed to be known in advance.

3.2. Network Model

A TSN network is modelled as a directed graph $\mathrm{G(V, E)}$ where $V = ES \cup SW$ and $E \subseteq V \times V$ denote the set of end-systems (ES) and switches (SW), and the edges connecting the nodes, respectively. All network components' clocks are assumed to be synchronized allowing the open and close events of the gates of the egress ports to operate in coordination. An edge between the nodes a and b in E represents a full-duplex physical link corresponding to two directional logical links, namely, [a,b] and [b,a]. Each physical link [a,b] is characterized by the 3-tuple of ($\mathrm{R}_{[a,b]}$, $\beta_{[a,b]}$, $\mathrm{w}_{[a,b]}$) where $\mathrm{R}_{[a,b]}$, $\beta_{[a,b]}$, $\mathrm{w}_{[a,b]}$ represent the link rate, propagation delay and weight of the link [a,b], respectively. The link rate has a direct impact on the transmission time such that the transmission time of a frame with length L to be transmitted over a link [a,b] with a rate of $\mathrm{R}_{[a,b]}$ can be formulated as $\frac{L}{R_{[a,b]}}$. In this thesis, the link rates for the overall network are assumed to be equal to 1 Gb/s. The propagation delay results from the physical properties of the transmission medium and the length of the transmission link. The weight of a link is a configuration parameter which has

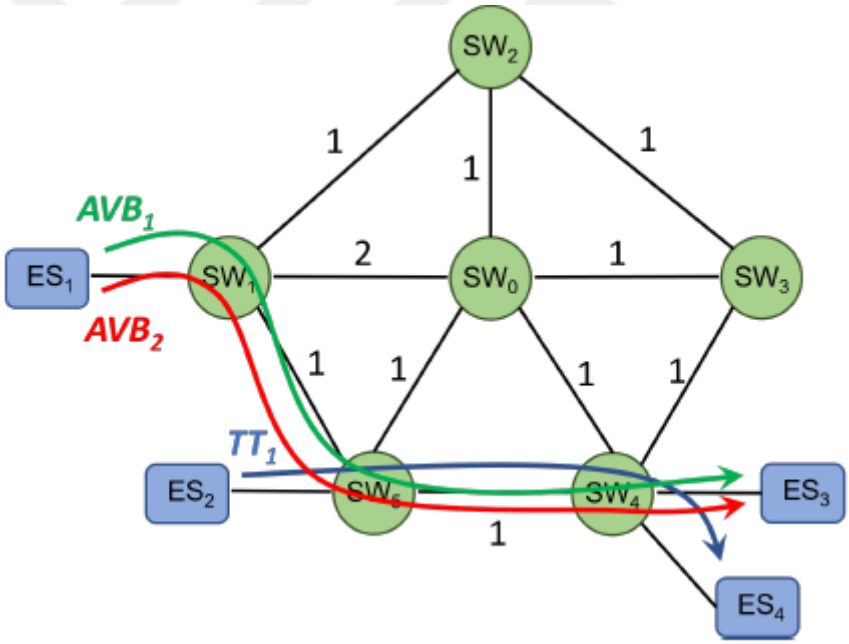
a direct impact on the computation of shortest transmission routes for the streams. In this work, each link in the physical topology is assumed to have a unit weight.

4. PROBLEM FORMULATION

In this thesis, we address the routing problem for AVB streams, which is known to be NP-Hard [6, 8]. The addressed problem has three inputs:

- A weighted graph of the physical topology $G(V, E)$,
- The set of AVB streams S_{AVB} to be transmitted in the network along with their timing requirements,
- The set of TT streams S_{TT} to be transmitted in the network along with their GCLs and routes given.

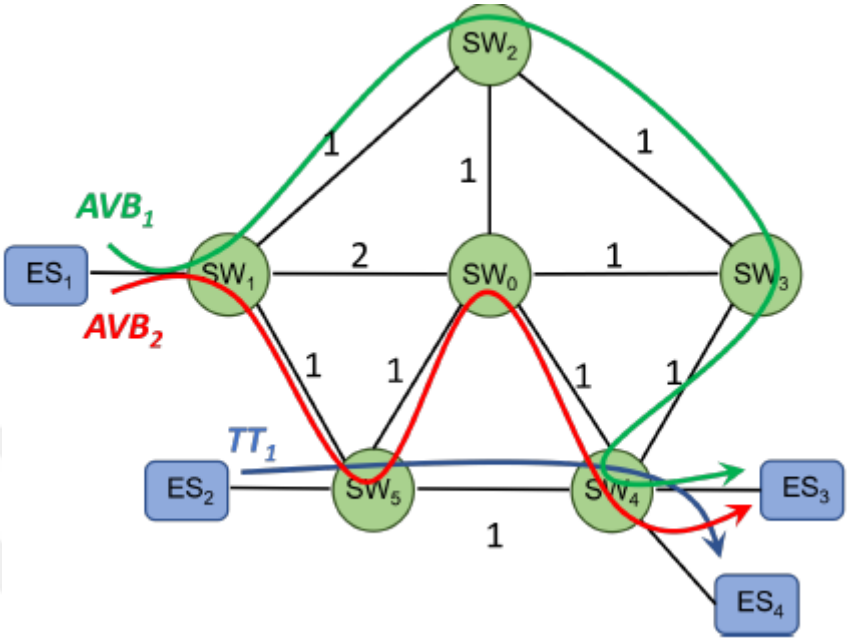
The output to the problem is a routing solution for AVB streams which satisfies their timing requirements**.**



(a) Shortest Routing

Figure 4.1. A motivational example for AVB routing problem

continuation of Figure 4.1.



(b) Alternative Routing

Figure 4.1. shows a motivational example for the AVB routing problem addressed in this thesis, which contains a physical topology with 4 ESs and 6 SWs. The traffic scenario for this example includes two AVB streams, namely AVB_1 and AVB_2, from ES_1 towards ES_3, and one TT stream, namely TT_1, from ES_2 towards ES_4 using the given route of $ES_2 \rightarrow SW_5 \rightarrow SW_4 \rightarrow ES_4$. Figure 4.1a shows a routing solution for AVB streams utilizing the shortest routes from ES_1 to ES_3 where AVB_1, AVB_2 and TT_1 share the common link of $SW_5 \rightarrow SW_4$. On the other hand, Figure 4.1b depicts an alternative routing solution where AVB_1 and AVB_2 are transmitted over longer routes so that a common link is no longer shared with TT_1. The superiority of the routing solutions presented in Figures 4.1a and 4.1b in terms of the schedulability of AVB streams depends on the amount of interference from TT_1, the interference of AVB_1 and AVB_2 to each other and the AVB route lengths. In this thesis, we propose an MTR-based optimization approach which computes candidate transmission routes with diverse characteristics for AVB streams, which have varying lengths and overlapping, and avoid the TT streams at varying levels.

5. MTR BASED TRAFFIC OPTIMIZATION

In this section, we present our MTR-based traffic optimization approach for AVB streams, namely TO_{MTR}, using GRASP meta-heuristic. In each iteration of GRASP, an initial solution is firstly constructed based on the MTR concept, and a local search is performed on the initial solution to achieve a local minimum. At the end of each iteration, if the cost of the achieved local minimum is lower than the cost of the best solution found so far, the local solution is stored as the new best one. GRASP-based optimization terminates if the elapsed time exceeds a given time limit.

In contrast to the existing research work solely relying on the physical topology to construct an initial solution for GRASP, TO_{MTR} reduces the search space for GRASP to consider only the K shortest paths for each AVB stream, half of which is computed using the physical topology while the other half is determined using a VT. TO_{MTR} constructs the VT by assigning a sufficiently high weight to each link in the VT if it is traversed by the route of a TT stream in order to exclude it from the AVB routes so that the interference from the TT streams is reduced. On the other hand, the AVB routes computed over the physical topology do not provide such a property. If there exists a large number of TT streams to be transmitted in the network, the routes from the physical topology are more likely to be shorter than the routes from the VT since the number of links in the VT available for data transmission becomes significantly smaller due to the existence of a high number of isolated links. Furthermore, depending on the source and destination nodes of an AVB stream, the richer path diversity provided by the physical topology in the case of a large number of TT streams may allow the AVB routes from the physical topology to overlap with each other with a smaller probability, and hence, may limit the inter-stream interference for AVB streams.

Since the amount of avoidance from the TT streams, the length of an AVB route, and the amount of overlapping with other AVB routes all have an impact on the WCD of an AVB stream, thanks to the MTR concept, we allow GRASP to perform a more efficient local search by including a combination of routes with varying levels of TT avoidance, length, and overlapping within the candidate routes of each AVB stream, which increases the quality of an initial solution.

5.1. Cost Function

Similar to [6, 8], the cost of a routing solution $\mathcal{R}$ is defined as:

$$cost(\mathcal{R}) = O_1(\mathcal{R}) \times W_1 + O_2(\mathcal{R}) \times W_2 + O_3(\mathcal{R}) \times W_3 \quad (5.1)$$

where $O_1(\mathcal{R})$, $O_2(\mathcal{R})$, and $O_3(\mathcal{R})$ represent the total number of unschedulable AVB streams not satisfying the deadline constraints, the total amount of worst-case delays of AVB streams and the number of unique network links traversed by all the AVB streams, respectively, for the routing solution $\mathcal{R}$. In Eq 5.1, the weight W_1 is set to a very large value to prioritize the schedulable solutions while W_2 and W_3 are selected to be significantly smaller. Please note that O_3 may allow shorter routes or multicast trees with late branching points to be preferred so that a smaller number of links in the overall network is utilized for AVB transmissions.

5.1.1. Isolating Link Weight

The same weight as in the physical topology is assigned to each link in a VT if it is not traversed by the route of any TT stream. Otherwise, the weight of a link is set to a very large value, namely w_{high}, which is selected to be sufficiently high to exclude the link from the shortest route of any AVB stream if possible [22]:

$$w_{high} = |E| \times w_{max} \quad (5.2)$$

where $|E|$ and w_{max} represent the total number of links in the network topology and the maximum of the normal weight values in the physical topology, respectively.

Propositon: Setting the weight of each link transmitting TT traffic to w_{high} in a VT guarantees that the shortest route of each AVB stream avoids the TT links as long as there exists a route in the VT for each AVB stream which does not contain any link with a weight of w_{high}.

Proof: Assume that the shortest route R_S of an AVB stream computed over the VT does contain at least one isolated link in the case that there exists an alternative path R_A in the VT for the same stream which does not contain any isolated link The total weight of R_S, namely $W(R_S)$ becomes $W(R_S) \geq w_{high}$ since R_S traverses at least one isolated link. On the other hand, $W(R_A) \leq (|E| - 1) \times w_{max}$ since R_A traverses a total of $|E| - 1$ links with a weight of w_{max} in the worst case, taking into account that there exists isolated links in the VT, and R_A does not contain any isolated link. Since $(|E| - 1) \times w_{max} < |E| \times w_{max}$, $W(R_A) < W(R_S)$ contradicting with the assumption that R_S is the shortest route.

5.1.2. Algorithm

Algorithm 1: Iterative algorithm for GRASP

Data:$G(V,E), S_{AVB}, S_{TT}, K, TIME\_LIMIT$
Result: *Optimized routing solution for AVB streams*

$PT \leftarrow G(V,E)$
$VT \leftarrow constructVirtualTopology(PT, S_{TT})$
$CR_{all} \leftarrow computeCandidateRoutes(PT, VT, S_{AVB}, K)$

$\mathcal{R}_{best-so-far} \leftarrow \emptyset$
while $elapsed\_time \leq TIME\_LIMIT$ **do**
  $\mathcal{R}_{ini} \leftarrow constructInitialSolution(CR_{all})$
  $\mathcal{R}_{local} \leftarrow performLocalSearch(\mathcal{R}_{ini})$

  if $cost(\mathcal{R}_{local}) < cost(\mathcal{R}_{best-so-far})$ **then**
    $\mathcal{R}_{best-so-far} \leftarrow \mathcal{R}_{local}$
  end
end

return $\mathcal{R}_{best-so-far}$

Algorithm 1 presents the iterative algorithm for TO_{MTR} which is inputted by the graph of the physical topology ($G(V,E)$), the sets of the AVB and TT streams to be transmitted in the network (S_{AVB} and S_{TT}) the number of candidate routes to be considered for each AVB stream (K) and the time limit for GRASP to terminate (TIME_LIMIT). Please note that the routing information of each TT stream is assumed to be known in advance and inputted to

Algorithm 1 as part of S_{TT}. The graph of the physical topology is stored within PT in line 1 while the corresponding VT is constructed by *constructVirtualTopology()* in line 2 based on the physical topology and S_{TT}. In line 3, K candidate routes are computed by *computeCandidateRoutes()* for each AVB stream using the Yen's algorithm [23], and stored within the data structure of CR_{all}. *constructInitialSolution()* implements the same randomized greedy algorithm as in [6, 8] in line 7 to compute the best initial solution by taking into the consideration the candidate routes in CR_{all}. A local search on $\mathcal{R}_{ini}$ is performed by *performLocalSearch()* in line 8 to find the local minimum solution $\mathcal{R}_{local}$, which is stored as the best solution found so far, namely $\mathcal{R}_{best-so-far}$, in line 11 if its cost is lower than the one of the previous best solution. The cost of a routing solution formulated in Eq 5.1 is determined with the help of the *AVB Latency Math* tool standardized in IEEE 802.1BA as implemented in [8] to evaluate the worst-case delay of each AVB stream. The loop in lines 6 to 13 continues to execute until the elapsed time exceeds the given time limit. $\mathcal{R}_{best-so-far}$ is returned as the ultimate optimum solution in line 15.

Algorithm 2: Virtual Topology Construction

```
   Data: G(V, E), S_TT
   Result: Constructed virtual topology
1  VT ← G(V, E)
2  for each s ∈ S_TT do
3      for each l ∈ R_s do
4          w(l)_VT ← w_high
5      end
6  end
7
8  return VT
```

Algorithm 2 shows the algorithm for *constructVirtualTopology()* in Algorithm 1 which constructs the VT iterating through each link in the VT traversed by the route R_s of each TT stream s to assign the weight of w_{high} as indicated by the lines 2 to 6, and returns the constructed VT in line 8. The time complexity of Algorithm 2 is bounded by $\mathcal{O}(|\mathrm{E}| \times |\mathrm{S_{TT}}|)$. In the case that every link in the network is used to transmit at least one TT stream, it is not possible for the AVB streams to totally avoid the TT streams in which case *constructVirtualTopology()* leads to a VT where all the links have a weight of w_{high}.

constructVirtualTopology() can be easily extended to cover such traffic scenarios by differentiating the network links transmitting a relatively higher volume of TT traffic from the remaining ones, and assigning w_{high} to such links while keeping the weight of the remaining links the same as the ones in the physical topology.

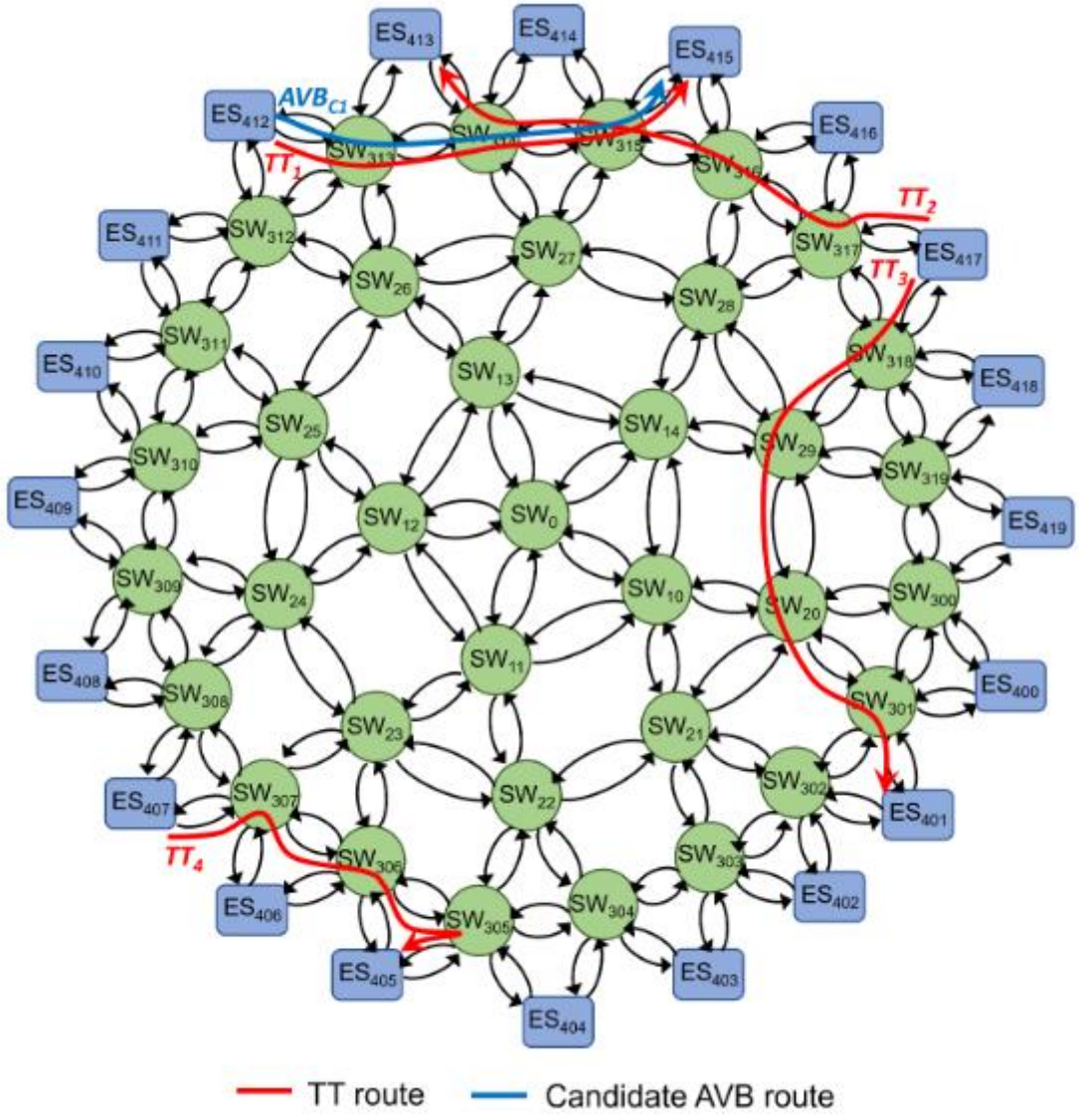
Algorithm 3: Computation of Candidate Routes

Data: $G(V,E), VT, S_{AVB}, K$
Result: *Candidate routes for each AVB stream*

1 $CR_{all} \leftarrow \emptyset$
2 **for *each*** $s \in S_{AVB}$ **do**
3 $CR_s \leftarrow \emptyset$
4 $R_{PHY} \leftarrow computeKShortestPaths(G(V,E), \lfloor K/2 \rfloor, s)$
5 $R_{VT} \leftarrow computeKShortestPaths(VT, \lceil K/2 \rceil, s)$
6
7 $CR_s \leftarrow CR_s \cup R_{PHY} \cup R_{VT}$
8 $CR_{all} \leftarrow CR_{all} \cup CR_s$
9 **end**
10
11 **return** CR_{all}

Algorithm 3 presents the algorithm for *computeCandidateRoutes()* in Algorithm 1 which computes the candidate routes for each AVB stream to be considered while constructing the initial solution for GRASP. Algorithm 3 iterates through each AVB stream to compute the corresponding candidate routes in lines 2 to 9. $\lfloor K/2 \rfloor$ shortest paths, namely R_{PHY}, for the current AVB stream are constructed by using the physical topology in line 4 via the Yen's algorithm while $\lceil K/2 \rceil$ shortest paths, namely R_{VT}, are computed using the VT in line 5. The time complexity of the Yen's algorithm is $\mathcal{O}\big(K|V|(|E| + |V|\log|V|)\big)$ [23]. The candidate routes computed for each AVB stream, namely CR_s, are stored in CR_{all} in line 8, and the resulting CR_{all} from all iterations is returned in line 11.

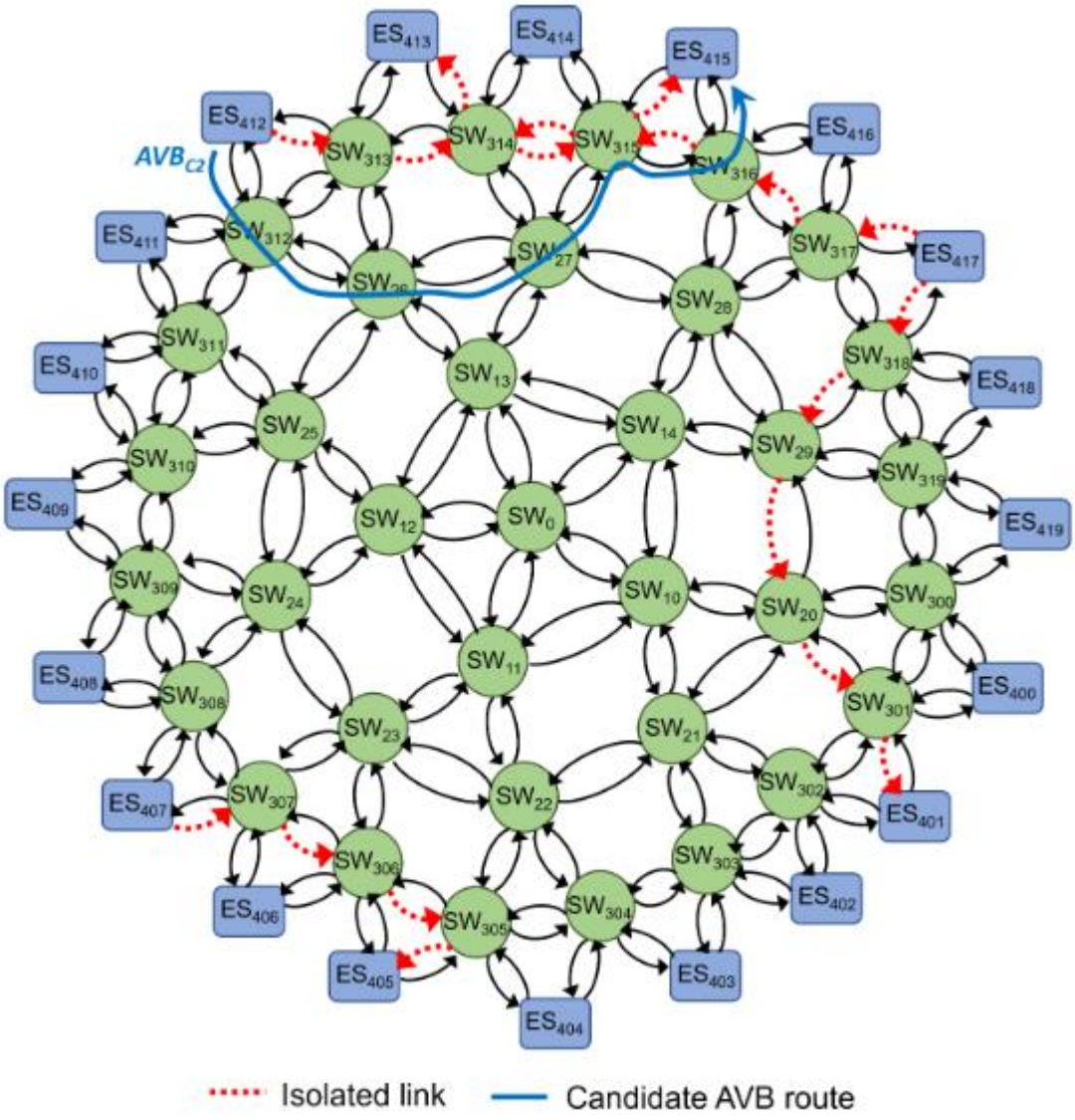
5.1.3. Operational Example



(a). Physical topology with example TT streams

Figure 5.1. Physical/virtual topologies for ABB network and candidate AVB routes computed by TO_{MTR} for K=2

continuation of Figure 5.1.



(b). Virtual topology

Figure 5.1a. shows the physical topology for the ABB network [35] with 20 ESs and 36 SWs over which four different TT streams, namely TT_1, TT_2, TT_3 and TT_4, are transmitted using the routes indicated by the red lines. Figure 5.1b. demonstrates the corresponding VT computed by *constructVirtualTopology()* of TO_{MTR} in Algorithm 2 using the physical topology and TT routes depicted in Figure 5.1a. where the links with a weight of w_{high} are indicated by the red dotted directional lines. For an AVB stream originating from ES_{412} and destined to ES_{415}, Figures 5.1a. and 5.1b. also demonstrates the 1st and 2nd shortest routes indicated by the blue lines, namely AVB_{C1} and AVB_{C2}, computed by

computeCandidateRoutes() of TO_{MTR} in Algorithm 3 for K=2. Please note that *computeCandidateRoutes()* runs the Yen's algorithm with K=1 sequentially over physical and virtual topologies to determine AVB_{C1} and AVB_{C2}, respectively, which share 4 and 0 common links with the TT streams shown in Figure 5.1a.

Table 5.1. Possibilities for 1st and 2nd shortest routes of an example AVB stream between ES_{412} and ES_{415} in case of K=2

Optimization Type	1st Shortest Route		2nd Shortest Route	
	Route	#TT Links	Potential Routes	#TT Links
TO_{PHY}	$ES_{412} \rightarrow SW_{313} \rightarrow SW_{314} \rightarrow SW_{315} \rightarrow ES_{415}$	4	$ES_{412} \rightarrow SW_{312} \rightarrow SW_{26} \rightarrow SW_{27} \rightarrow SW_{315} \rightarrow ES_{415}$	1
			$ES_{412} \rightarrow SW_{313} \rightarrow SW_{26} \rightarrow SW_{27} \rightarrow SW_{315} \rightarrow ES_{415}$	2
			$ES_{412} \rightarrow SW_{313} \rightarrow SW_{314} \rightarrow SW_{27} \rightarrow SW_{315} \rightarrow ES_{415}$	3
			$ES_{412} \rightarrow SW_{313} \rightarrow SW_{314} \rightarrow SW_{315} \rightarrow SW_{316} \rightarrow ES_{415}$	3
TO_{MTR}	$ES_{412} \rightarrow SW_{313} \rightarrow SW_{314} \rightarrow SW_{315} \rightarrow ES_{415}$	4	$ES_{412} \rightarrow SW_{312} \rightarrow SW_{26} \rightarrow SW_{27} \rightarrow SW_{315} \rightarrow SW_{316} \rightarrow ES_{415}$	0

On the other hand, Table 5.1. presents the 1st and 2nd shortest routes computed by the conventional traffic optimization approach, namely TO_{PHY}, for K=2 which computes all the candidate routes for an AVB stream using solely the physical topology. While the 1st shortest route is the same for TO_{PHY} and TO_{MTR}, Yen's algorithm computes four different possibilities for the 2nd shortest route in case of TO_{PHY}, each of which shares at least one common link with the TT streams in Figure 5.1a. This operational example shows that TO_{MTR} is more likely to achieve candidate routes totally avoiding the TT streams in addition to the shortest ones leading to an enhanced diversity of candidate routes for the initial solutions.

6. EXPERIMENTAL RESULTS

In this section, we implement TO_{MTR} approach proposed in Section 5.1.2. by extending the traffic optimization implementation from [8], namely TO_{PHY}, which computes all the candidate routes for a stream using the physical topology. We evaluate the performance of TO_{MTR} by constructing realistic unicast and multicast traffic scenarios with varying number of AVB and TT streams. Our scenarios rely on the ABB network topology due to its rich path diversity for which TO_{MTR} constructs the virtual topologies shown in Section 5. In our experiments, we analyze the impact of different K values on end-to-end delays by using the *AVB Latency Math* tool specified by 802.1BA. We compare the performance of TO_{MTR}, with Dijkstra and TO_{PHY} in terms of the number of unschedulable AVB streams (O_1) which do not meet the specified deadline constraints, the improvement percentage (IP) by which TO_{MTR} reduces O_1 resulting from TO_{PHY}, and the number of unique network links traversed by all the AVB streams (O_3). We also report experimental results to provide a reasoning why TO_{MTR} achieves a lower O_1 compared to TO_{PHY} in most of the experiments. IP value achieved by TO_{MTR}, for a certain scenario is formulated as follows:

$$IP = \frac{(\theta - \varphi) \times 100}{\theta} \tag{6.1}$$

where θ and φ denote the O_1 values resulted from TO_{PHY} and TO_{MTR}, respectively. All the experiments in this section are conducted using a PC with Intel i7-8750H processor running at 2.20 GHz and 16 GB RAM.

6.1. Traffic Scenarios

For the experiments, we construct two different sets of traffic scenarios which contain unicast and multicast data transmissions by extending the scenarios provided in [8]. The first set includes 7 scenarios, namely U_1 to U_7, where unicast transmissions take place while the second set includes 5 scenarios, namely M_1 to M_5, for multicast transmissions. The number of AVB streams to be transmitted in the network is varied as 20, 35 and 50 for U_1 to U_7 while it is fixed to 16 for M_1 to M_5. On the other hand, the number of TT streams is

varied between 6 and 16 for both the unicast and multicast scenarios. To be able to exploit the GCLs provided by [8], the routes of the TT streams are selected to be non-overlapping with each other in all the scenarios so that the number of distinct links in overall network which transmit a TT stream increases in proportion to S_{TT}. All the attributes for each AVB and TT stream including message payload size, message period, number of frames per period and deadline are determined in the same way as in [8]. Similar to [8] the payload size is randomly selected from the range of 11-400 bytes for AVB messages while the number of AVB frames transmitted per period is arbitrarily chosen from the range of 1-10. In all the scenarios, the link rates for the overall network are assumed to be equal to 1 Gb/s.

6.1.1. Performance Evaluation

Table 6.1. Performance evaluation for unicast scenarios

Scenario Name	$\|S_{AVB}\|$	$\|S_{TT}\|$	*Dijkstra*		TO_{PHY}		TO_{MTR}		*IP*
			O_1	O_3	O_1	O_3	O_1	O_3	
U_1	35	6	9	189	8	189	6	189	25
U_2		8	12	189	11	189	7	192	36.36
U_3		10	16	189	15	189	9	195	40
U_4		13	18	189	17	189	10	195	41.17
U_5		16	19	189	18	186	13	196	27.77
U_6	20		13	114	13	115	7	120	46.15
U_7	50		31	274	29	274	24	287	17.24

(a) K = 2

Scenario Name	TO_{PHY}		TO_{MTR}		*IP*
	O_1	O_3	O_1	O_3	
U_1	7	189	6	189	14.28
U_2	10	191	6	193	40
U_3	15	192	8	196	46.66
U_4	16	191	9	196	43.75
U_5	18	191	11	198	38.88
U_6	12	117	6	119	50
U_7	28	278	22	289	21.42

continuation of Table 6.1

(b) K = 3

Scenario Name	TO_{PHY}		TO_{MTR}		*IP*
	O_1	O_3	O_1	O_3	
U_1	4	190	6	189	-50
U_2	6	193	6	193	0
U_3	11	194	8	196	27.27
U_4	13	194	9	196	30.76
U_5	16	193	10	197	37.50
U_6	11	117	6	119	45.45
U_7	26	281	22	284	15.38

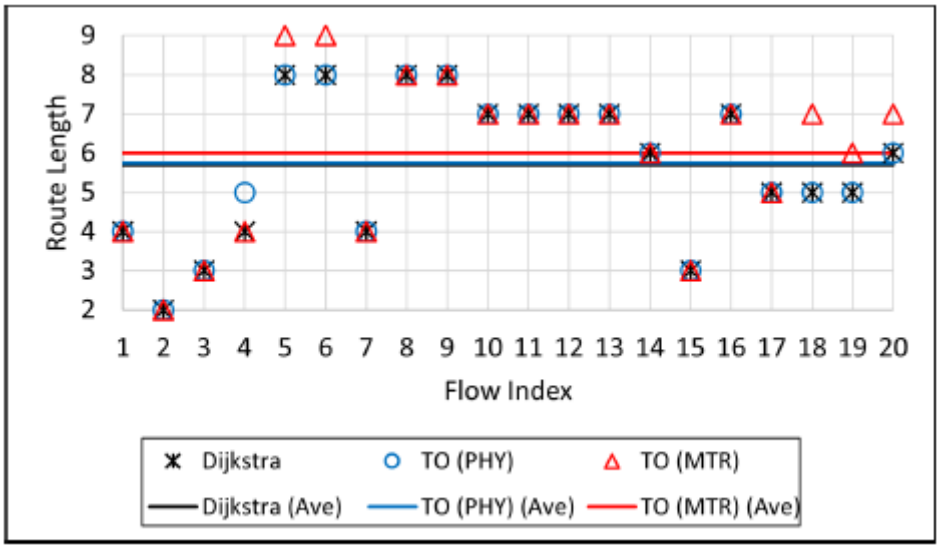
(c) K = 4

Scenario Name	TO_{PHY}		TO_{MTR}		*IP*
	O_1	O_3	O_1	O_3	
U_1	2	194	1	198	50
U_2	4	196	2	199	50
U_3	6	198	5	199	16.66
U_4	8	197	6	199	25
U_5	11	199	10	201	9.09
U_6	6	121	5	121	16.66
U_7	22	284	19	290	13.63

(d) K = 50

Table 6.1. shows the performance results with respect to small values of K using the unicast scenarios for TO_{MTR}, TO_{PHY} and Dijkstra, which always uses the shortest routes. For each scenario in this experiment, GRASP is executed for a total of 3 minutes, which was empirically verified to be largely sufficient to find stabilized solutions since the cost value is no longer improved by GRASP after this time limit. The respective numbers of AVB and TT streams ($|S_{TT}|$ and $|S_{AVB}|$) transmitted in each scenario are reported in Table 6.1a Similarly, the performance results for Dijkstra are reported in Table 6.1a along with the results for TO_{MTR} and TO_{PHY} for K=2 but omitted from Table 6.1b and 6.1c since it operates independent of K. As shown in the table, TO_{MTR} achieves a significantly lower O_1 in almost all the scenarios. We believe that it is an important scalability result to attain low O_1 even for the small values of K since it reduces the time complexity of GRASP to find good-quality

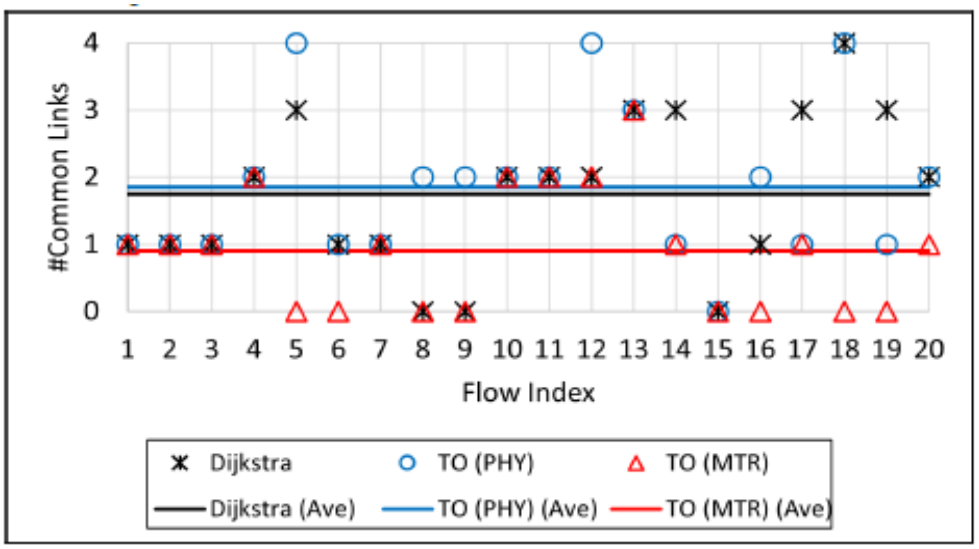
solutions especially for the large problem instances. For example, in case of U_6 with K=2, O_1 varies as 13, 13 and 7 for Dijkstra, TO_{PHY} and TO_{MTR}, respectively, which corresponds to an IP value of 46.15%. Table 6.1. also shows that TO_{MTR} requires a slightly higher number of links to be utilized in overall network for AVB transmissions in almost all the scenarios as indicated by O_3. For example, in case of U_6 with K=2, O_3 varies as 114, 115 and 120 for Dijkstra, TO_{PHY} and TO_{MTR}, respectively.



(a) Route Lengths

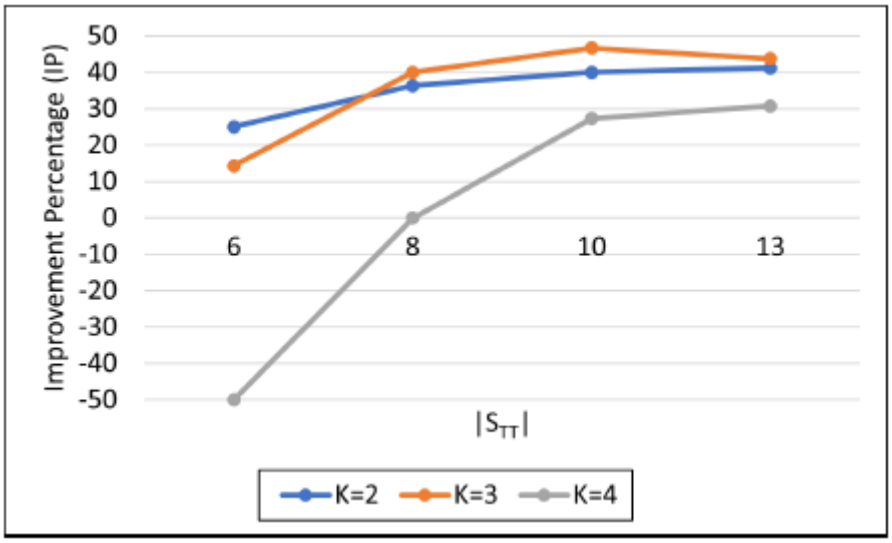
Figure 6.1. AVB route lengths and the number of common links share AVB and TT streams for the scenario of U_6 in Table 6.1a

continuation of Figure 6.1

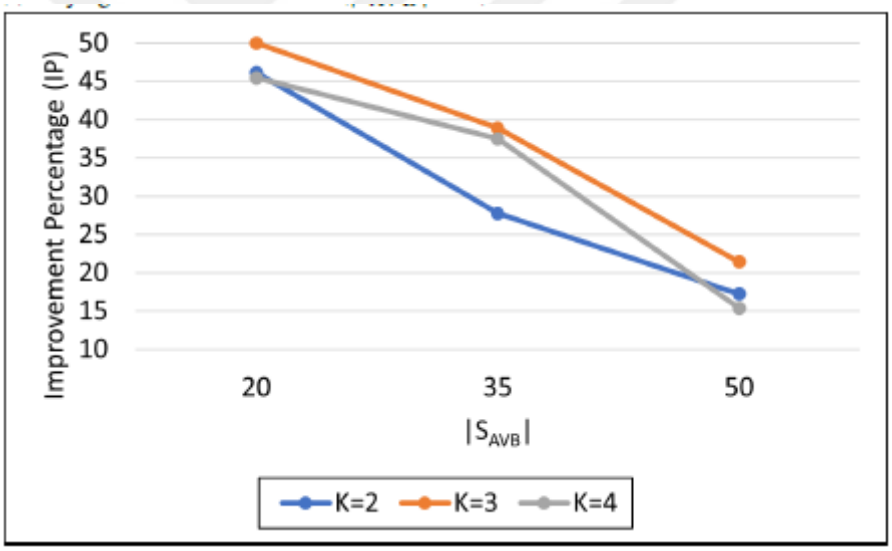


(b) Common Links

Figure 6.1. depicts the value distributions for the AVB route lengths and the number of common links shared by the AVB and TT streams for the scenario of U_6 with K=2, which contains a total of 20 and 16 AVB and TT streams, respectively. Figure 6.1a demonstrates that the AVB route length on average varies as 5.7, 5.75 and 6 for Dijkstra, TO_{PHY} and TO_{MTR}, respectively. On the other hand, Figure 6.1b shows that the number of links on the AVB routes which are also used to transmit TT streams varies as 1.75, 1.85 and 0.9 on average for Dijkstra, TO_{PHY} and TO_{MTR} respectively. These results indicate that TO_{MTR} may lead to slightly longer AVB routes in order to better avoid the TT streams which in turn results in lower O_1 values due to an improved network latency compared to other approaches. Please note that TO_{MTR} yields $O_1 = 7$ for U_6 with K = 2 while the other approaches result in $O_1 = 13$.



(a). Varying number of TT streams $|S_{AVB}| = 35$)



(b). Varying number of AVB streams $|S_{TT}| = 16$)

Figure 6.2. Improvement percentages (IP) with respect to varying number of streams

Figure 6.2. shows the improvement percentages achieved by TO_{MTR} with respect to varying number of streams transmitted in the network. Figure 6.2a demonstrates the effect of varying number of TT streams, namely 6, 8, 10 and 13, which correspond to the scenarios of U_1 to U_4, respectively, where $|S_{AVB}| = 35$. As shown in Figure 6.2a, IP tends to increase with respect to $|S_{TT}|$ up to a saturation point, which is caused by the fact that it becomes increasingly more difficult for the AVB routes to avoid the TT streams since a higher number

of common links are more likely shared by the AVB and TT streams as $|S_{TT}|$ increases. For example, IP values in case of K=3 vary as 14.28%, 40%, 46.66% and 43.75% for $|S_{TT}|$ = 6 to $|S_{TT}|$ = 13, respectively. Please note that, after the saturation point, the amount of improvement achieved by TO_{MTR} may be inevitably degraded since the network topology may no longer provide the sufficient level of path diversity to avoid the TT streams. Figure 6.2.b. reports the IP values with respect to varying number of AVB streams, namely 20, 35 and 50, which correspond to the scenarios of U_6, U_5 and U_7, respectively, where $|S_{TT}|$ = 16. As shown in Figure 6.2.b., IP tends to decrease with respect to $|S_{AVB}|$ because the amount of interference among the AVB streams increases since the network links are more likely shared by a higher number of AVB streams as $|S_{AVB}|$ increases. For example, IP values in case of K=3 vary as 50%, 38.88% and 21.42% for $|S_{AVB}|$ = 20, 35 and 50, respectively. Table 6.1d shows the performance results for a large value of K, namely K = 50, using the unicast scenarios. For each scenario in this experiment, GRASP is executed for a total of 10 minutes, which was empirically verified to be sufficient to stabilize a solution. As shown in the table, TO_{MTR} achieves a lower O_1 in all the cases while mostly resulting in a slightly higher number of links to be utilized. However, the IP values are smaller in general compared to the values reported in Table 6.1 for the small values of K. For example, the IP value achieved for U_5 is observed to be 9.09% while it varies as 27.77%, 38.88% and 37.5% for K=2, K=3 and K=4, respectively. This is due to the fact that GRASP considers a larger number of candidate routes for each stream for the larger values of K which results in better investigating the search space of the problem domain, and, hence, increasing the probability of finding a higher-quality solution for both TO_{MTR} and TO_{PHY}. Still, the lower O_1 values achieved by TO_{MTR} as reported in Table 6.1d indicate that the quality of the candidate routes in case of TO_{MTR} is higher than in the case of TO_{PHY}.

Table 6.2. Performance evaluation for multicast scenarios

Scenario Name	$\|S_{AVB}\|$	$\|S_{TT}\|$	*Dijkstra*		TO_{PHY}		TO_{MTR}		*IP*
			O_1	O_3	O_1	O_3	O_1	O_3	
M_1	16	6	6	163	6	160	4	158	33.33
M_2		8	8	163	8	160	4	159	50
M_3		10	10	163	9	160	4	157	55.55
M_4		13	10	163	10	160	5	156	50
M_5		16	11	163	10	160	6	156	40

(a) K = 2

Scenario Name	TO_{PHY}		TO_{MTR}		*IP*
	O_1	O_3	O_1	O_3	
M_1	5	158	4	158	20
M_2	7	157	4	159	42.85
M_3	9	158	4	157	55.55
M_4	9	160	5	156	44.44
M_5	10	158	6	156	40

(b) K = 3

Scenario Name	TO_{PHY}		TO_{MTR}		*IP*
	O_1	O_3	O_1	O_3	
M_1	4	150	2	155	50
M_2	5	151	2	158	60
M_3	7	155	4	156	42.85
M_4	7	152	5	159	28.57
M_5	8	151	6	157	25

(c) K = 4

Scenario Name	TO_{PHY}		TO_{MTR}		*IP*
	O_1	O_3	O_1	O_3	
M_1	0	148	1	146	$-\infty$
M_2	2	148	1	150	50
M_3	3	150	3	149	0
M_4	5	146	4	150	20
M_5	6	150	5	154	16.66

(d) K = 50

Table 6.2. shows the performance results with respect to small values of K using the multicast scenarios. Similar to the experiments in Table 6.1. GRASP is executed for a total of 3 minutes for each multicast scenario. As shown in the table, TO_{MTR} achieves a lower O_1 in all the scenarios. For example, in case of M_3 with K=3, O_1 varies as 10, 9 and 4 for Dijkstra, TO_{PHY} and TO_{MTR}, respectively, which corresponds to an IP value of 55.55%. In contrast to the O_3 values in Table 6.1a, TO_{MTR} leads to a smaller number of links to be utilized for AVB multicast transmissions especially for K=2 compared to Dijkstra and TO_{PHY}. For example, O_3 varies as 163, 160 and 156 in case of M_5 with K=2 for Dijkstra, TO_{PHY} and TO_{MTR}, respectively. Table 6.2d shows the performance results for K=50 using the multicast scenarios. Similar to the experiments in Table 6.1d, GRASP is executed for a total of 10 minutes for each scenario. As shown in the table, TO_{MTR} achieves a lower O_1 in most of the cases. The IP values in Table 6.2d exhibit a similar behavior as in Table 6.1d by being smaller in general compared to the values reported in Table 6.1. for the small values of K. For example, the IP value achieved for M_4 is observed to be 20% while it varies as 50%, 44.44% and 28.57% for K=2, K=3 and K=4, respectively.

7. CONCLUSION

IEEE 802.1 TSN task group is the leading organization which targets the standardization of Ethernet-based deterministic real-time communication technologies. In this thesis, a MTR based optimization approach is developed for the route planning of AVB streams in order to satisfy their timing requirements in a TSN network with mixed-criticality support. Our approach computes candidate routes to be used to construct initial solutions for GRASP which searches for the best routing solution. Thanks to the diverse forwarding capabilities provided by the MTR concept, we allow GRASP to perform a more efficient local search by including a combination of routes with diverse characteristics within the candidates leading to a good-quality solution. We evaluate the performance of our approach by constructing realistic unicast and multicast traffic scenarios with varying number of AVB and TT streams. Experimental results show that our approach significantly reduces the number of unschedulable AVB streams in almost all the scenarios compared to the other approaches in the literature.

8. RECOMMENDATIONS

In this thesis, a combination of the physical topology and a single VT is used to compute the candidate routes for each AVB stream to be evaluated by the GRASP meta-heuristic to find the best solution. This methodology relying on the MTR concept allows the candidate routes to have more diverse characteristics, namely varying levels of length, overlapping and TT avoidance, compared to the sole utilization of the physical topology for route planning, leading to an increased likelihood for GRASP to find a better solution. However, we believe that the diversity among the candidate routes can be further improved by intelligently constructing more than one VT, which in turn may enhance the quality of the solution. Since optimization approach itself has a direct impact on the effectiveness of the solution to the routing problem, we also plan to apply our MTR based search space reduction technique to construct initial solutions for other meta-heuristic approaches such as Tabu search [36] and genetic algorithms [37] in order to solve the routing problem for AVB streams. We also believe that route planning for TT and UBS streams is another potential field of research to which our MTR based approach can be applied. In this thesis, AVB Latency Math is used for the delay analysis of AVB streams. However, the tool presented in [38] relying on Network Calculus is considered to be integrated into our evaluation as a future work in order to enhance the accuracy of the AVB delay analysis. Furthermore, a delay analysis is planned to be accomplished through network simulation by developing an OMNeT++ [39] simulation environment for TSN-related experiments.

9. REFERENCES

1. Neumann, P., Communication in industrial automation—What is going on?, Control Engineering Practice, 15,11 (2007) 1332-1347.

2. https://www.sae.org/standards/content/as6802 Time-Triggered Ethernet AS6802. 6 June 2022

3. He, Q., Zeng, Q. and Tang, X., Research and implement on industry control networks based on embedded SERCOS-III protocol, Electronics, Communications and Control (ICECC), 2021.

4. Kleines, H., Detert, S., Drochner, M. and Suxdorf, F., Performance Aspects of PROFINET IO, 2007 15th IEEE-NPSS Real-Time Conference, April 2007, USA, 1 – 5.

5. Raagard, M., L., Pop, P., Optimization algorithms for the scheduling of IEEE 802.1 Time-Sensitive Networking (TSN), DTU Technical Report, Technical University of Denmark, 2017

6. Pop, P., Raagard, M., L., Craciunas, S. and Steiner, W., Design Optimization of Cyber-Physical Distributed Systems using IEEE Time-sensitive Networks (TSN), IET Cyber-Physical Systems: Theory & Applications, 1 (2016).

7. Gavrilut, V., Zhao, L., Raagard, M., L. and Pop, P., AVB-Aware Routing and Scheduling of Time-Triggered Traffic for TSN, IEEE Access, 6 (2018) 75229-75243.

8. Laursen, S., M., Pop, P. and Steiner, W., Routing Optimization of AVB Streams in TSN Networks, SIGBED Rev., 13,4 (2016) 43–48.

9. IEEE Standard for Local and metropolitan area networks--Audio Video Bridging (AVB) Systems, IEEE Std 802.1BA-2021 (Revision of IEEE Std 802.1BA-2011), (2011) 1 – 45.

10. https://1.ieee802.org/tsn/ Time-Sensitive Networking (TSN) Task Group. 30 May 2022.

11. Zara, C., Routing Algorithms for Real-Time Traffic on IEEE 802.1 Time-Sensitive Networking, Master Thesis, Department of Applied Mathematics and Computer Science, Denmark, 2018.

12. IEEE Standard for Local and Metropolitan Area Networks–Timing and Synchronization for Time-Sensitive Applications, IEEE Std 802.1AS-2020 (Revision of IEEE Std 802.1AS-2011), (2020) 1 – 421.

13. Smith Jr., M., E., Evaluation of IEEE 802.1 Time Sensitive Networking Performance for Microgrid and Smart Grid Power System Applications, Master Thesis, University of Tennessee, USA, 2018

14. Ditzel, G., Time Sensitive Network (TSN) Protocols and use in EtherNet/IP Systems, 2015 ODVA Industry Conference & 17th Annual Meeting, October 2015, USA.

15. Lo Bello, L., Steiner, W., A Perspective on IEEE Time-Sensitive Networking for Industrial Communication and Automation Systems, Proceedings of the IEEE, 107,6 (2019) 1094 - 1120.

16. Danielis, P., Parzyjegla, H., Mühl, G., Schweißguth**,** E. and Timmerman, D., Frame Replication and Elimination for Reliability in Time-Sensitive Networks, CoRR, abs/2109.13677 (2021).

17. Ergenç, D., Fischer, M., Implementation and Orchestration of IEEE 802.1CB FRER in OMNeT++, 2021 IEEE International Conference on Communications Workshops (ICC Workshops), 2021, 1-6.

18. IEEE Standard for Local and Metropolitan Area Networks–Bridges and Bridged Networks – Amendment 31: Stream Reservation Protocol (SRP) Enhancements and Performance Improvements, IEEE Std 802.1Qcc-2018 (Amendment to IEEE Std 802.1Q-2018 as amended by IEEE Std 802.1Qcp-2018), (2018) 1 – 208.

19. Psenak, P., Mirtorabi, S., Roy, A., Nguen, L., and Pillay-Esnault, P., MT - OSPF: Multi topology (MT) routing in OSPF, RFC 4915, (2007).

20. Przygienda, T., Sagl, Z., Shen, N. and Sheth, N., M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs), RFC 5120, (2008).

21. Bae, S., Handerson, T., R., Traffic Engineering with OSPF Multi-Topology Routing, MILCOM 2007 - IEEE Military Communications Conference, October 2007, USA, 1 – 7.

22. Kvalbein, A., Hansen, A., F., Cicic, T., Gjessing, S. and Lysne O., Multiple Routing Configurations for Fast IP Network Recovery, IEEE/ACM Transactions on Networking, 17,2 (2009) 473 – 486.

23. Yen, J., Y., Finding the K Shortest Loopless Paths in a Network, Management Science, 17,11 (1971) 712–716.

24. Chudakov, S., YenKShortestPath https://jgrapht.org/javadoc/ org.jgrapht.core/org/jgrapht/alg/shortestpath/YenKShortestPath.html. 2 June 2022.

25. https://en.wikipedia.org/wiki/Yen%27s_algorithm Yen's Algorithm. 2 June 2022.

26. Eppstein, D., Finding the k Shortest Paths, SIAM J. Comput., 28,2 (1999) 652–673.

27. Chudakov, S., EppsteinKShortestPath https://jgrapht.org/javadoc/ org.jgrapht.core/org/jgrapht/alg/shortestpath/EppsteinKShortestPath.html 2 June 2022.

28. Atallah, A., A., Hamad, G., B. and Otmane, A., Fault-Resilient Topology Planning and Traffic Configuration for IEEE 802.1Qbv TSN Networks, 2018 IEEE 24th International Symposium on On-Line Testing and Robust System Design (IOLTS), July 2018, Spain, 151 – 156.

29. Gavrilut, V., Zarrin, B., Pop, P. and Samii, S., Fault-Tolerant Topology and Routing Synthesis for IEEE Time-Sensitive Networking, Proceedings of the 25th International Conference on Real-Time Networks and Systems, October 2017, France, 267 – 276.

30. Huang, K., Wan, X., Wang, K., Jiang, X., Chen, J., Deng, Q., Xu, W., Peng, Y. and Liu, Z., Reliability-Aware Multipath Routing of Time-Triggered Traffic in Time-Sensitive Networks, Electronics, 10 (2021) 125.

31. Nayak, N., G., Dürr, F. and Rothermel, K., Routing Algorithms for IEEE802.1Qbv Networks, SIGBED Rev., 15,3 (2018) 13 – 18.

32. Schweißguth**,** E., Danielis, P., Timmermann, D., Parzyjegla, H. and Mühl, G., ILP-Based Joint Routing and Scheduling for Time-Triggered Networks, Proceedings of the 25th International Conference on Real-Time Networks and Systems, October 2017, France, 8 – 17.

33. Ojewale, M. A., Yomsi, P., M., Routing Heuristics for Load-Balanced Transmission in TSN-Based Networks, SIGBED Rev., 16,4 (2020) 20 – 25.

34. Kvalbein, A., Lysne, O., Robust Load Balancing Using Multi-Topology Routing, 1 – 6

35. https://global.abb/group/en ABB Group. Leading digital technologies for industry — ABB Group. 24 May 2022.

36. Dürr, F., Nayak, N., G., No-Wait Packet Scheduling for IEEE Time-Sensitive Networks (TSN), Proceedings of the 24th International Conference on Real-Time Networks and Systems, October 2016, France, 203 – 212.

37. Li, X., Huang, N. and Fei, Z., A genetic algorithm based configuration optimization method for AFDX, 2014 10th International Conference on Reliability, Maintainability and Safety (ICRMS), August 2014, China, 440 – 444.

38. Zhao, L., Pop, P., Zheng, Z., Daigmorte, H. and Boyer, M., Latency Analysis of Multiple Classes of AVB Traffic in TSN with Standard Credit Behavior using Network Calculus, CoRR, abs/2005.08256 (2020).

39. https://omnetpp.org/ OMNeT++ Discrete Event Simulator. 22 May 2022

CIRRICULUM VITAE

He graduated from the Hamdi Bozbağ High School in 2014. He received his B.Sc. from the Department of Computer Engineering at Karadeniz Technical University, Trabzon, Turkey in 2020. From 2020 up to present, he has been studying as a student in the MSc program of the Department of Computer Engineering at Karadeniz Technical University. His research area includes MTR and IEEE 802.1 TSN routing.

1. Demir, Ö., K., Cevher, S., Multi-Topology Routing Based Traffic Optimization for IEEE 802.1 Time Sensitive Networking, Real-Time Syst, 2022, (Submitted)