

Received July 8, 2021, accepted July 22, 2021, date of publication July 28, 2021, date of current version August 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3101021

# Simulator for Performance Evaluation of ASON/GMPLS Network

**SYLWESTER KACZMAREK<sup>1</sup>** AND **MAGDALENA MŁYNARCZUK<sup>1</sup>**

Faculty of Electronics, Telecommunications, and Informatics, Gdansk University of Technology, 80233 Gdansk, Poland

Corresponding author: Sylwester Kaczmarek (kasy1@eti.pg.edu.pl)

**ABSTRACT** The hierarchical control plane network architecture of Automatically Switched Optical Network with utilization of Generalized Multi-Protocol Label Switching protocols is compliant to next generation networks requirements and can supply connections with required quality of service, even with incomplete domain information. Considering connection control, connection management and network management, the controllers of this architecture could perform the same operations on the transport plane resources and could support control functions for Software Defined Networking controllers, which are considered as the future networking solutions. Therefore, it is very important to determine factors that have an influence on service control and resource management. One of the tools for achieving this goal is computer simulation. The paper proposes a simulator for the hierarchical control plane networks in an open source OMNeT++ environment, in order to evaluate network architecture performance for different network structures and traffic parameters. To demonstrate simulator capabilities, examples of typical simulations performance results for Polish and European network structures are presented and discussed. Performance metrics which have been used to evaluate the performance of this architecture are: mean values of Call Set-up Time, Connection Set-up Time, Call Release Time, Connection Release Time, loss probability for inter-domain connection requests and loss probability for low and high priority requests. General remarks concerned with presented simulator are also provided.

**INDEX TERMS** ASON, GMPLS, hierarchical control plane, OMNeT++ simulator, quality of service.

## I. INTRODUCTION

The Automatically Switched Optical Network (ASON) is a standardized network architecture proposed by International Telecommunication Union - Telecommunication Standardization Sector (ITU-T) in [1]. The main purpose of ASON control plane is to facilitate fast and efficient configuration of connections within a transport layer network to support both switched and soft permanent connections in a high capacity core optical network. The hierarchical control plane approach, which is presented in [1], gives the opportunity to control multidomain network under the condition of incomplete domain information and can satisfy current and future needs of network operator for supplying services with strictly defined Quality of Service (QoS). The ASON control plane is composed of different components, which provide specific functions with the use of routing and signaling protocols. The protocol suite that can potentially

be used to implement the general control plane function is Generalized Multi-Protocol Label Switching (GMPLS) [2], [3]. The ASON approach utilizing GMPLS protocols by control plane is named as ASON/GMPLS. Actually, ASON recommendations are in many areas compliant with Software Defined Networking (SDN) concept. Common control aspects for ASON and SDN controllers are presented in [4], [5]. The first laboratory trial tests for ASON/GMPLS and SDN were conducted and are presented in [6].

For the commercial success, both SDN and ASON/GMPLS architectures with hierarchical control plane structure should be properly designed. Therefore, we should examine and determine which parameters have significant impact on ASON/GMPLS control plane performance. Because of the complexity of analytical models for optical network architectures, which map all ASON/GMPLS functions, the simulation method is an effective technique for evaluating control plane architecture's performance. Many factors can affect network performance. Because of that, utilization of simulators for performance evaluation and examination

The associate editor coordinating the review of this manuscript and approving it for publication was Guangjie Han<sup>1</sup>.

TABLE 1. Comparison of simulators.

	ASON/GMPLS simulator	Asons simulator	IP/MPLS over ASON/GMPLS simulator	Viena LTE-A simulator	MASEMUL
Is the simulator built in an open-source environment?	+	+	+	---	---
Is the simulator written in a general-purpose programming language?	+	+	+	+	+
Is the simulator modular designed?	+	+	+	+	+
Is designed for control of optical network?	+	+	+	---	---
Is designed for hierarchical ASON/GMPLS network architecture?	+	---	---	---	---
Is designed for core network simulation?	+	+	+	---	---
Can users perform simulations for different structures of networks?	+	+	+	+	+
Have users the ability to set and service different network service classes/priorities?	+	---	+	---	+
Is it possible to configure traffic sources, traffic classes, transport resources	+	+	+	+	+

of dependencies are highly desirable for research purpose. Conducting evaluations in the simulated environment allow examination of ASON/GMPLS performance, without the need to carry out experiments on the actual or future installed network.

Significant work has been completed for simulators of IP/MPLS over ASON/GMPLS networks. The exemplary results for a flat control plane architecture are presented in [7]. Apart from [7], we propose the simulator with the hierarchical ASON/GMPLS approach with a control plane implementation including Resource Reservation Protocol with Traffic Engineering Extensions (RSVP-TE) [8], Diameter protocol [9], Link Management Protocol (LMP) [10] and Open Shortest Path First (OSPF) mechanisms protocols [11]. The functionality and certain general description of the presented simulator without implementation details could be also found in [12], [13]. In [12], we compared the efficiency of flat to hierarchical control plane of ASON/GMPLS structures. In [13], we have examined in a simulation, two resource reservation algorithms to indicate, whether they have significant impact on required quality of service in ASON/GMPLS network with hierarchical control plane structure.

Currently, simulation is the basic tool for researching telecommunications networks due to the complexity of the systems and solved problems for both mobile and stationary users. The examples for the first are Viena Long Term Evolution-Advanced (LTE-A) simulator [14] and MANET Agent-based Simulator for Evaluating communications with MULTimedia (MASEMUL) [15]. Our considerations are focused on the ASON/GMPLS with hierarchical control plane structure networks for stationary users.

Tab. 1 compares the most relevant features of telecommunications network simulators, in the context of this work. The comparison includes ASON/GMPLS simulator presented in the paper, Asons simulator [16], IP/MPLS over ASON/GMPLS simulator [7], Viena LTE-A simulator [14], and MASEMUL [15]. The main advantage of our

ASON/GMPLS simulator is its ability to demonstrate and simulate optical hierarchical control plane functionalities with differentiated service quality classes. The presented simulator has a modular design based on compound modules. Due to this, it can be easily expanded in order to examine different resource reservation algorithms, protection and restoration mechanisms, and budget QoS parameters. Just like Asons and IP/MPLS over ASON/GMPLS simulator, it is suitable for simulating core optical control plane network. Apart from Viena Long Term Evolution-Advanced (LTE-A) simulator, which is MATLAB based, ASON/GMPLS does not require a license for certain toolboxes.

In the paper our contributions to development of ASON/GMPLS network architecture are:

- design and implementation of ASON/GMPLS network simulator with hierarchical control plane structure in an open-source OMNeT++ environment,
- mapping control elements functions and protocols mechanisms in the simulator,
- proposition and implementation of communication between hierarchical control plane structures with Diameter protocol and LMP procedures which are not included in general ITU-T recommendations,
- presentation simulator capabilities by its simulation results.

The paper is organized as follows. The standardization of ASON/GMPLS architecture is presented in section II. Section III is devoted to the structure of the simulator, a recommended call service ASON/GMPLS scenario and a validation process. In section IV, exemplary results for simulation model for Polish structure and simulation model for European structure are presented and discussed. The paper is summarized in section V.

## II. ASON/GMPLS HIERARCHICAL ARCHITECTURE

ASON hierarchical control plane standardization, initiated by ITU-T primarily defines the components of the control plane and interactions between these components. The control

plane is represented by call control components and connection control components.

The node consists of the Connection Controller (CC), the Routing Controller (RC) and the Link Resource Manager (LRM) for a single level in a routing area hierarchy. View of the status and utilization of resources of the transport plane is provided to the LRM by the Termination and Adaptation Performer (TAP). The Protocol Controller (PC) supports the function of mapping the parameters of the abstract interfaces of the control components into protocol messages. In the case of hierarchical approach, connection controllers are related to one another in a hierarchical manner and could play the role of Network Representant, Domain Representant or Node. General remarks and guidelines for architecture and control plane components are presented in [1], [2], [11], [8]. Additionally, the Internet Engineering Task Force (IETF) has defined the extensions for signaling and routing protocols. The requirements for Generalized MPLS (GMPLS) protocols usage and extensions for ASON can also be found in [17], [18], [19].

Despite the standardization, the recommendations for hierarchical ASON/GMPLS routing are not very detailed. Method of hierarchical QoS routing based on a network resource reservation and algorithms for hierarchical routing could be found in [20], [21]. Multi-domain ASON/GMPLS network operations for hierarchical routing are presented in [22]. Simulators for hierarchical ASON/GMPLS with detailed architecture control plane implementation are not available. Most of the research in this area is focused on hierarchical routing algorithms without implementation details for ASON/GMPLS network. Advancement of the works which have been carried out for control aspects for ASON indicates that ASON/GMPLS hierarchical control plane structure is sufficient for supporting SDN controller [23], [24]. Due to this, the authors propose a simulator for hierarchical control plane performance.

In order to examine functionality of the ASON/GMPLS architecture, the simulator implementation was preceded by the implementation of the functionality of the ASON/GMPLS architecture in the laboratory at Gdansk University of Technology. The results of work within the ASON/GMPLS testbed network are presented in [25]. The solutions implemented in the laboratory testbed were transferred to the simulator to examine their effectiveness for more complex network topologies.

### III. ASON/GMPLS SIMULATOR

The simulator models ASON/GMPLS architecture with hierarchical control plane (for signaling messages) and the transport (data) plane. The signaling messages from the control plane are handled by the transport plane consisting of Optical Cross-Connects (OXCs) with Dense Wavelength Division Multiplexing (DWDM) optical links characterized by the physical length and capacity (bandwidth) at each wavelength. The two lowest layers of the Open Systems Interconnection model (OSI model) with no signal parameters – except for bit

rate, message transmission time, including the Ethernet frame header and propagation delay – are simulated in the transport layer.

In the control plane, all higher layers of the OSI model resulting from the handling of ASON/GMPLS architecture protocols together with their header lengths, are simulated taking into account their impact on transmission time. At the top of the stack, are the applications responsible for control of the call service and scenarios.

The ASON/GMPLS simulator is realized in OMNeT++ environment, which is an object-oriented modular discrete event network simulation framework developed by András Varga [26].

The architecture of OMNeT++ simulations with ASON/GMPLS functional blocks is presented in Fig. 1. The blocks in grey represent the OMNeT++ environment components as follows [27]:

- simulation kernel and class library (*SIM*),
- Model Component Library with simple module definitions, their C++ implementations, compound modules types, channels, queues and messages types,
- library with code common to all user interfaces (*ENVIR*),
- envir-based libraries that contain specific user interface implementations (*Cmdenv*, *Tkenv*, *Qtenv*).

The ASON/GMPLS simulator consists of six main functional blocks: control plane, transport plane, call generation, topology and resource configuration, initial configuration, measurements which are integrated in OMNeT++ executable architecture. An implementation case for European network divided into three domains (represented as blue, green and pink nodes) based on ASON/GMPLS simulator is presented in Fig. 2.

The control plane consists of functional elements for call and connection functions. Call control functions are represented by Calling/Called Party Controllers (CCC1, CCC2), a Network Call Controller (NCC) and additional element an *ids*, which is responsible for call identifier assignment and call release generation.

The main messages of the *ids* are: *get\_freecallid*, *freecallid*, *busycallid*, *callconfirmed*, *releaseconfirmed*.

In the simulator the connection control functions are performed by nodes named as Control Elements (CEs).

Each Control Element consists of the Connection Controller (CC), the Routing Controller (RC) and the Link Resource Manager (LRM). The Termination and Adaptation Performer (TAP) function and interaction with the LRM is provided in *lrm* module. The RC functions and its interaction with CC are provided in *cc* module. Each CE provides an interface to OXC configuration and PC functions. The number of the CE is equal to the number of nodes. The example of CE structure for European network is presented in Fig. 3.

In hierarchical control plane, the hierarchy Connection Controller could play the role of CE Network Representant, CE Domain Representant or CE Node.

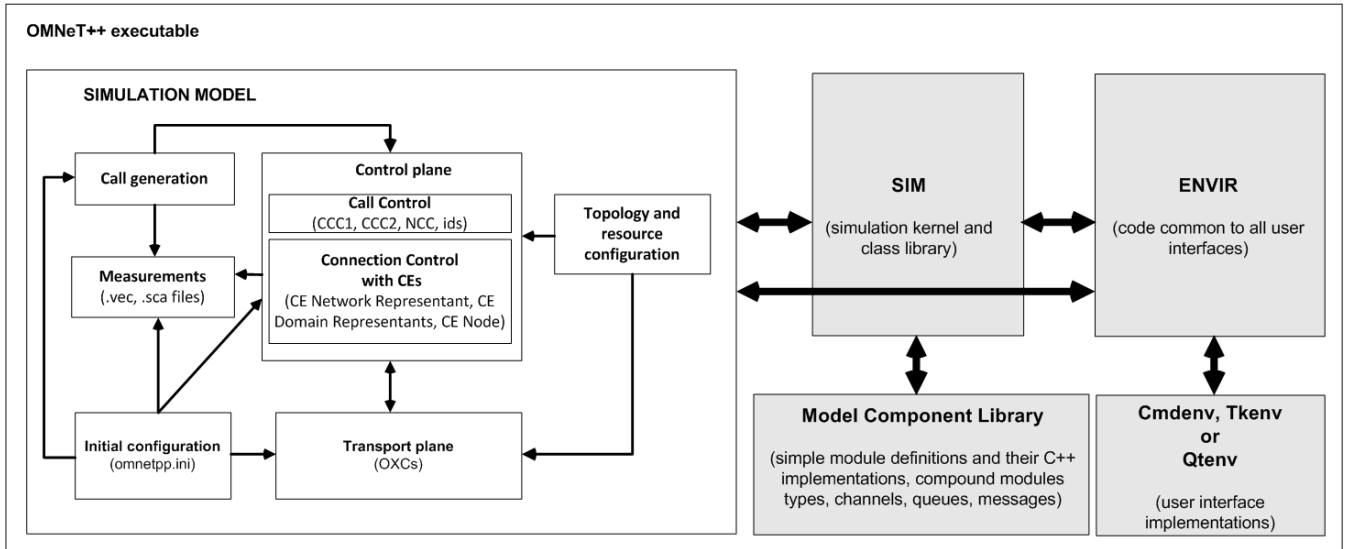


FIGURE 1. OMNeT++ executable architecture with ASON/GMPLS functional blocks.

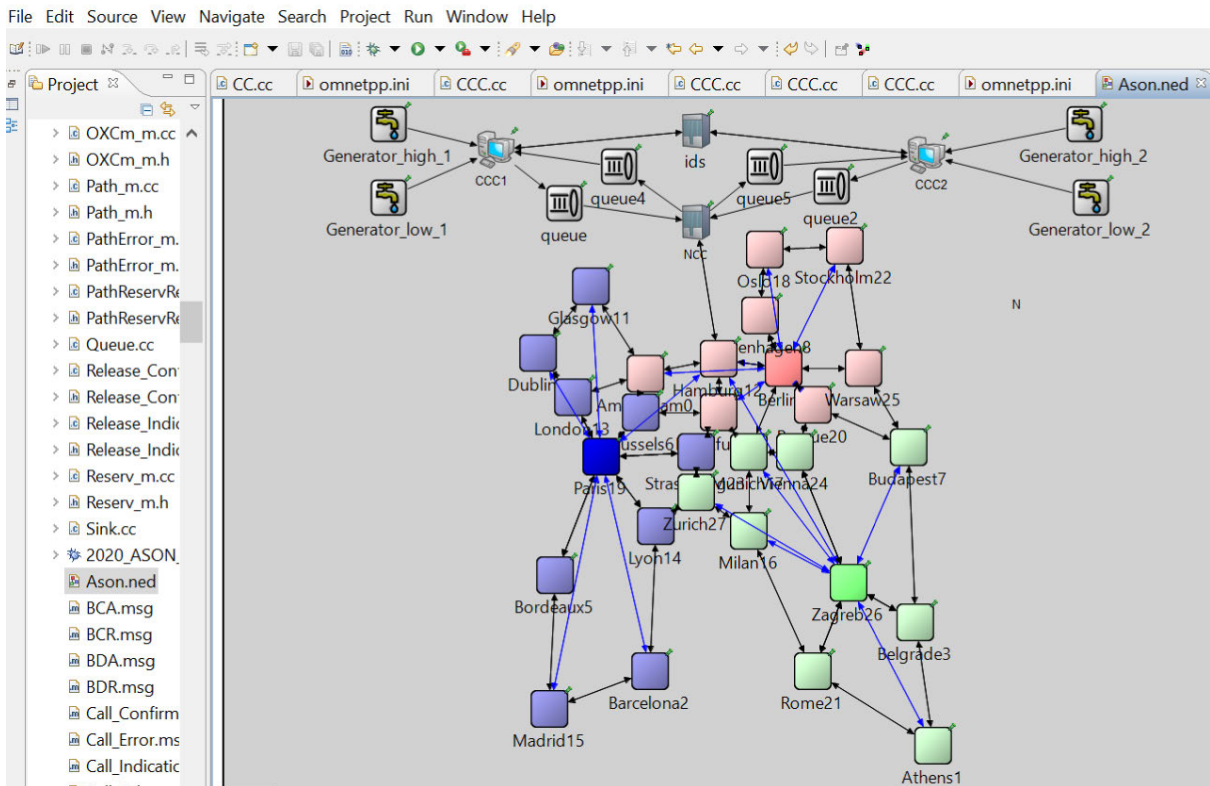


FIGURE 2. The ASON/GMPLS simulator for European structure.

The CE Network Representant and CE Domain Representant are related to one another in a hierarchical manner. Due to this each domain routing area has its own Connection Controller (CC) that has knowledge of the topology of its routing area but has no knowledge of the topology of routing areas above or below itself in the hierarchy, or other routing areas

at the same level in the hierarchy. On each level of hierarchy configurations for LRM and RC are different according to ITU-T recommendations.

The transport plane block simulates Optical Cross-Connects (OXCs) operations. A value of blocking probability is assumed for each OXC block. Signaling is performed on

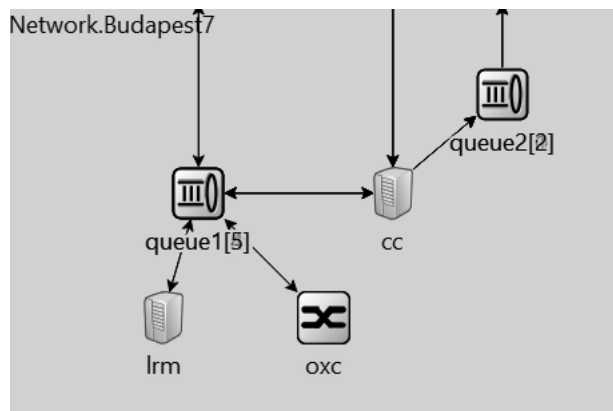


FIGURE 3. The structure of the control element (CE).

a separate wavelength. The resource allocation takes into consideration Routing and Wavelength Assignment problem (RWA) and connection control functions are performed in order to supply a wavelength continuity constraint condition.

Call generation is performed by generators module with default Mersenne Twister RNG (MT) presented in Fig. 2 as Generator\_high\_1, Generator\_high\_2, Generator\_low\_1, Generator\_low\_2. The call requests are generated according to request distribution defined in configuration file named as *omnetpp.ini* and request matrix in module *CCC1* and *CCC2* for low and high priority requests.

Topology and resource configuration is performed based on dedicated additional scripts in Python language for SNDlib [28] native format which generate network structure description (as *.ned* file) and Link Resource Manager configuration. It gives the model a great flexibility in regards to different network structures examination.

An initial configuration is additionally implemented in *omnetpp.ini* file (detailed and presented in further part of the section), where CE Network Representant, CE Domain Representants or CE Nodes for each domain are established. The ASON/GMPLS simulator enables different inter-domain configurations.

Functional block measurement is performed in the *ids* module and NCC controller. Call and connection times are calculated based on results recorded by *ids*. Loss probabilities are calculated based on results recorded by the NCC. Call control functions concerned with call processing and connection control functions are responsible for set-up and release connections in the transport plane. The call control plane is not aware of transport plane topology and separation for call control addresses and connection control addresses is introduced.

Optical resource reservation in the connection control plane is performed with the Resource Reservation Protocol – Traffic Engineering (RSVP-TE) and standardized RSVP-TE messages [8]. The communication between Network Representant and Domain Representants, and Domain Representants and CE Nodes is realized with the use of Diameter protocol [9]. Routing functions are implemented

with Open Shortest Path First – Traffic Engineering (OSPF-TE) based on routing protocol mechanisms [11].

Each functional element on each output port has a separate service queue system in the simulator, with the infinite First In First Out queue (FIFO). Messages' sending time depends on the size of the message and signaling link capacity.

In the simulation system we can distinguish groups of components: files with functional elements implementation in C++ code, message definitions, an initial configuration file and a file with a network structure.

The implementation of functional elements of control plane and transport plane is performed in C++ files with *.h/.cc* suffixes. Call control elements and connection control elements are defined as *cSimpleModules* and combined in hierarchical manner. Each C++ file containing the implementation of the functionality of the model element has an identical structure in which we can distinguish: class definition, initialization function, *handleMessage()* function and *finish()* function (optional). The *handleMessage()* function is called for every message that arrives at the module. The *finish()* function is called, when simulation results are put into file via output vector and output scalars. Output vectors are time series data, recorded from simple modules, while output scalars are summary results, computed during the simulation and written out when the simulation completes.

In the presented simulator functionality of the RC and the TAP are implemented in *cc.cc* file and *lrm.cc* file. Messages are introduced in the simulator as *.msg* files with defined message type and data fields. The *omnetpp.ini* configuration file contains settings that control how the simulation is executed and values for model parameters.

The main parameters set in the *omnetpp.ini* configuration file for ASON/GMPLS simulator are: *sim-time-limit*, *warmup-period*, *scalar-recording*, *vector-recording*, *request distribution*, *numberOfSources*, *bandwidths for call requests*, *id for nodes*, *id for domains*, *oxc.blockingPropability*, *numberOfCallIds*, *elementaryBandwidth*, *waveBandwidth*, *protectedBandwidth*, *oxc.blockingPropability*, *BorderNodes*, *NetworkRepresentant*, *DomainsRepresentants*. Network structure is presented in Network Description (NED) language (as *.ned* file) that describes the module network structure with delay parameters and gates.

When simulation program is built, the messages files are translated into C++ code using the *opp\_msgc* program. Then all C++ sources are compiled and linked with the simulation kernel and a user interface library to form a simulation executable library. The *.ned* files are loaded dynamically in their original text forms when the simulation program starts. After simulation, the results are recorded in separate *.sca* and *.vci* files and analyzed by using dedicated scripts written in Python language.

#### A. MESSAGE FLOW FOR CALL SET-UP REQUEST

The ASON/GMPLS simulator provides call set-up, connection set-up, call-release, connection-release scenarios

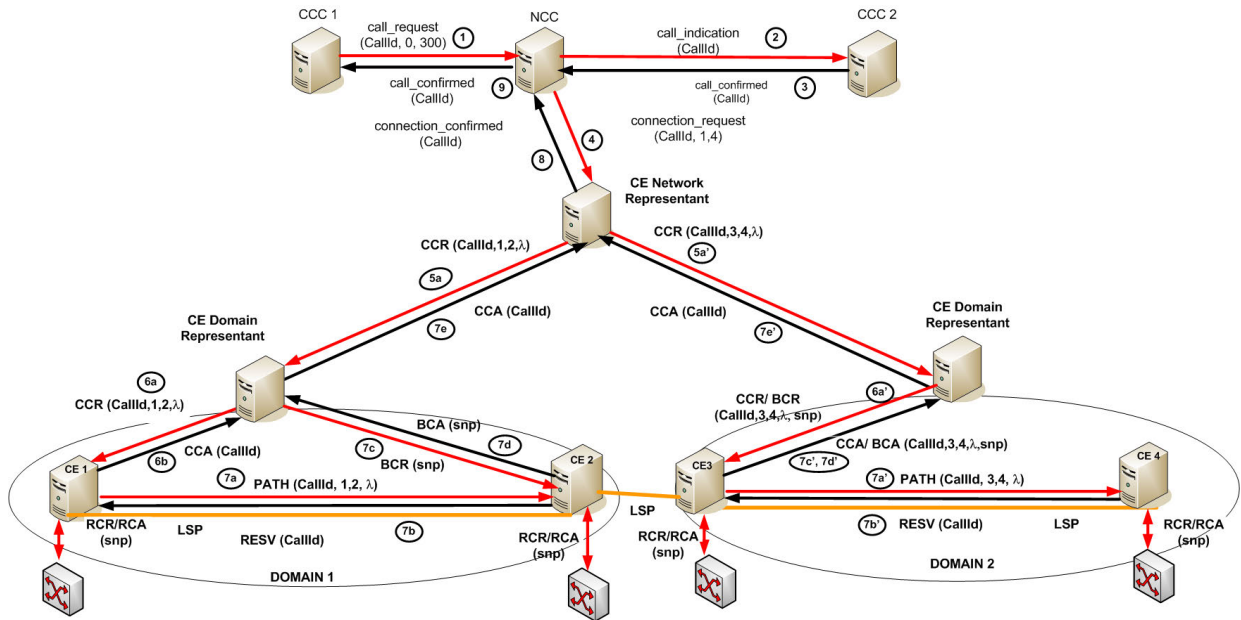


FIGURE 4. Call set-up scenario for ASON/GMPLS simulator.

in accordance with standardization recommendation for ASON/GMPLS network.

Due to the sheer size of conducted research, we have decided to focus in the paper only a successful call set-up scenario performed in accordance with standardization recommendation for ASON/GMPLS network. The message flow and Label Switching Path (LSP) establishment are presented in Fig. 4 and Fig. 5.

- 1) The Calling Party Call Controller (CCC1) initiates the submission of call request by sending the *call\_request* message to the Network Call Controller (NCC). The call message contains: the call identifier (CallId) assigned by block *ids*, the addresses of the calling party and the address of called party. The request also defines the required bandwidth and priority of the request.
- 2) NCC translates call source and destination identifiers to transport resource identifiers and informs the CCC2 about the occurrence of a request by sending a *call\_indication* message with the CallId number assigned.
- 3) The Called Party Controller (CCC2) acknowledges receipt of the service request message by sending *call\_confirmed*.
- 4) The NCC begins setting up the connection in the connection control plane by sending a *connection\_request* to the CE Network Representant. In the *connection\_request* message, the source node address and the destination node address of the connection control plane are transferred.
- 5) After receiving the message *connection\_request*, the CE Network Representant sends a query to its Routing Controller and creates Connection-Create-Request

(CCR) messages based on abstract view of domains, which are then send to Domains' Representants. Each CCR message consists of call identifier (CallId), and a pair of SubNetwork Points (SNPs) at the edge of the top level routing area. Additionally, required bandwidth, priority of request and wavelength are sent. The inter-domain link is chosen with uniform distribution.

- 6) The Domain Representant of Domain 1 sends CCR to the source node and Border-Create-Request (BCR) message to the edge node with inter-domain link SNP identifiers. The Domain Representant of Domain 2 sends corresponding CCR and BCR with inter-domain link SNP identifiers to the node of Domian 2.
- 7) After receiving routing information from the RC controller, the CE node of source Domain 1 sends a *lrm\_path* to the LRM for initial occupation of resources. After a positive reservation, the LRM returns *lrm\_response* message. The selected wavelength is passed in the *PATH* message to the next node on the connection path. A Resource-Create-Request (RCR) represented as *oxc\_configure* message is sent to the OXC optical switch. If the CE node is the last element, the answer is *lrm\_reserv\_pass*. Finally *oxc\_configure* (RCR) is sent to OXC and *RESV* message in order to confirm the reservation of resources in the transport plane. After successfully establishing the connection in the connection control plane and receiving the reservation confirmation at the edge node in form of Border-Create-Answer (BCA) message, the source node of Domain 1 (CE1) sends the Connection-Create-Answer (CCA) message to its Domain

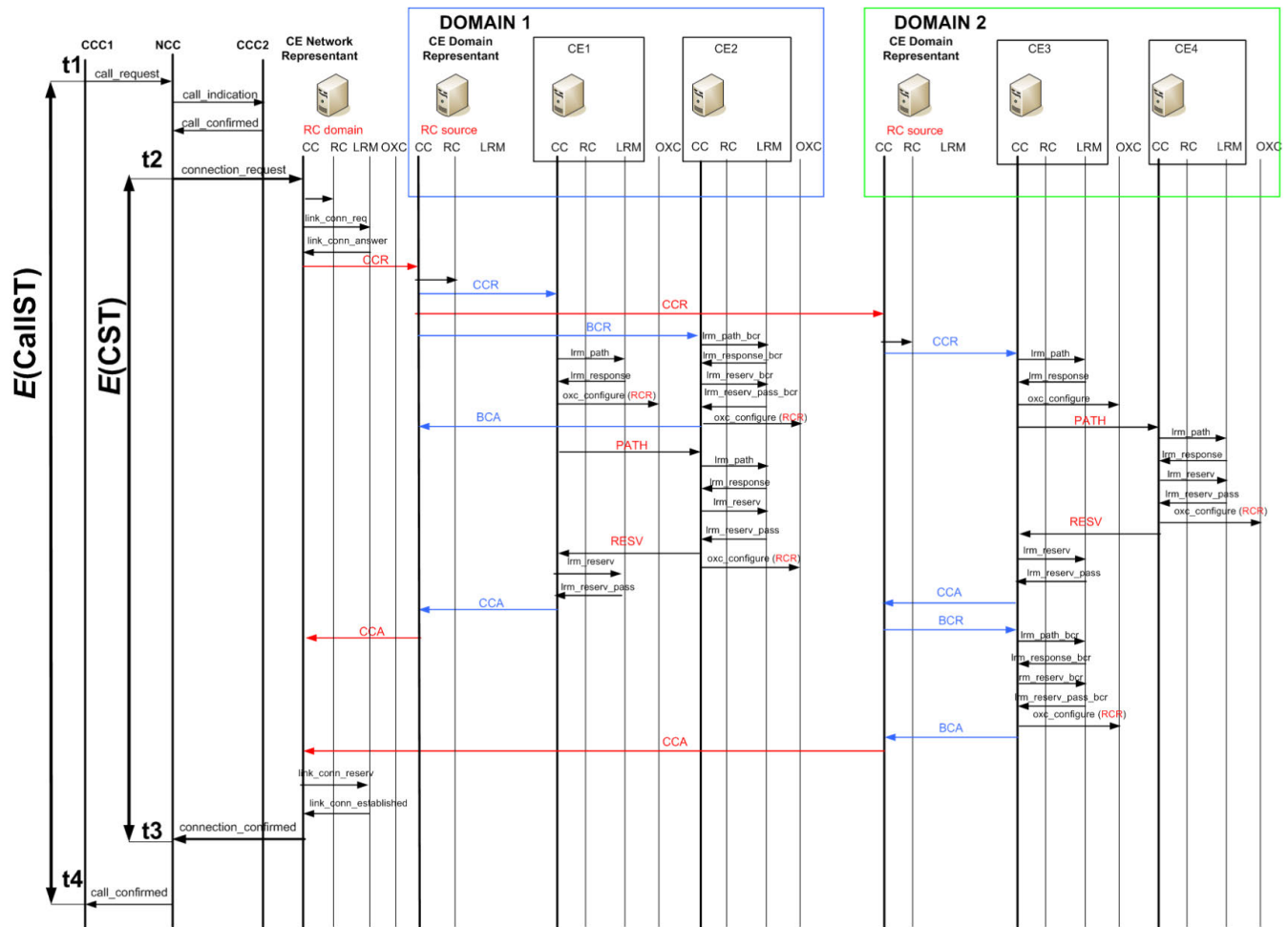


FIGURE 5. Message flow for call set-up scenario.

Representant. Corresponding RSVP-TE and Diameter messages are sent in Domain 2.

- 8) After receiving confirmation messages from Domains' Representants, the Network Representant sends *connection\_confirmed* message to NCC.
- 9) NCC informs the initiating CCC1 about the termination of the request by sending *call\_confirmed* with the given CallId.

Successfully establishment of LSP connection path is finalized with an appropriate *call\_confirmed* message, which is sent to the *ids* module.

The process of call release is invoked when the *ids* module send *busycallid* message and is not presented in the paper.

The Link Management Protocol (LMP) mechanisms are also provided in the simulator. Remote and local identifiers are mapped according to the Link Property Correlation procedure [10]. The examples of simulation results performed in the presented simulator are provided in the next section.

The simulator was examined in accordance with the recommendations. Verification and validation were conducted during the development of the simulator. Performed verification process confirmed that control plane scenarios are correctly

implemented and are compliant with ASON/GMPLS specifications.

Information about the ASON/GMPLS simulator are also available on Gdansk University of Technology's online platform "Most Wiedzy", accessed via WWW address <https://mostwiedzy.pl/pl/sylwester-kaczmarek,505-1/ason-gmpls-simulator> (last accessed 07/2021).

#### IV. SIMULATION RESULTS

In this section we demonstrate expanded simulation results to illustrate the variety of simulations options and the simulator capabilities for different datasets and network structures. Using the simulator many sets of input variables can be changed. Amongst them are traffic source parameters, network structures and resource reservation algorithms.

We are capable of examining the impact of various factors on processing time and loss probabilities.

We performed simulations with the use of simulation models for Polish and European structures. Both are based on Survivable Network Design Library (SNDlib) [28]. The Polish and European structures consist of respectively 12 and 28 nodes. Each structure has been divided into three

level hierarchies with pool resource reservation algorithm approach presented in [29].

The simulations were performed with the following parameters:

- exponential distribution of call request, exponential distribution of connection release requests,
- mean connection duration time ( $E(t)$ ): 2 minutes, 5 minutes, 15 minutes,
- the percentage of all generated requests ( $\lambda_c$ ) for high priority (Ph):10%, 20%,
- inter-domain traffic (MD):20%, 30%,
- the percentage of resources on each link for protected pool for high priority requests (Pz):20%, 40%,
- blocking probability of OXC: 0.001,
- signaling link capacity:1Gb/s,
- wavelength capacity:1Gb/s,
- capacity of single connection requests in bandwidth units:1BU = 5Mb/s, 2BU = 10Mb/s, 3BU = 15Mb/s.

In the simulation model, the resources are divided into two sets: a protected resources pool for high priority connection requests ( $Pz = 20\%$  or  $Pz = 40\%$ ) and a common pool for low and high priority requests. The protected pool is used, if it is not possible to service high priority request using the common pool.

The total simulation time was 3600s with warm-up period 200s. The number of measurement intervals was 5. Obtained results (5 measurements intervals) were estimated using  $t$ -Student distribution with a confidence level equals 0.95.

We evaluate mean values of call and connections times: Call Set-up Time  $E(CallST)$  which is defined as the time from sending *call\_request* ( $t1$ ) up to *call\_confirmed* ( $t4$ ), mean Connection Set-up Time  $E(CST)$  which is defined as the time from sending *connection\_request* ( $t2$ ) up to *connection\_confirmed* ( $t3$ ) (Fig. 5), mean Call Release Time  $E(CallRT)$  which is defined as the time from sending *call\_release* up to *release\_confirmed* and mean Connection Release Time  $E(CRT)$  which is defined as *connection\_release* up to *connection\_release\_confirmed* (here not presented in Fig. 5).

Additionally, we examine loss probability for low priority requests ( $B_0$ ) defined as the ratio between the number of low lost priority requests and the total number of generated low priority requests and loss probability for high priority requests ( $B_1$ ) defined as the ratio between the number of high lost priority requests and the total number of generated high priority requests. Moreover, we evaluate loss probability for inter-domain connections requests ( $B_{MD}$ ) calculated for low and high priority connection requests as a ratio between lost low/high priority inter-domain connection requests and the total number of low/high priority inter-domain connection requests. The obtained results are presented in Fig. 6 - Fig. 12 for Polish structure and in Fig. 13 – Fig. 19 for European structure.

Other performance parameters that can be obtained by the simulator are  $E(CallST)$  for unsuccessful and successful call set-up request,  $E(CST)$  for unsuccessful and successful

connection set-up request. Additionally, we can examine loss probability for low priority requests ( $B_0$ ) and high priority request ( $B_1$ ) with distinction between the possible reasons of loss (lack of resources, blockade of OXC). The same approach is done for ( $B_{SD}$ ) and ( $B_{MD}$ ). The list of the parameters could be expanded. Depending on the collected results, it is possible to measure not only the response times of individual functional modules, but e.g. the probability of service request loss, connection at the level of the entire network or domain (operator). Due to the fact that simulations results are collected and processed off-line, the user can prepare external scripts e. g. in Python language and analyze data according to needs.

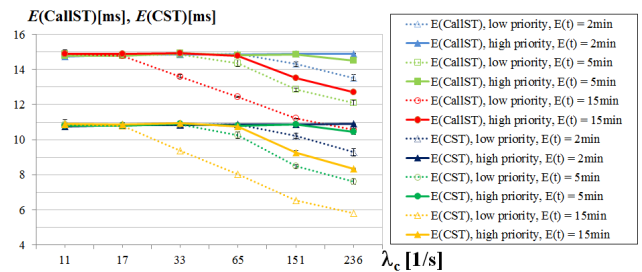


FIGURE 6. Mean Call Set-up Time and Connection Set-up Time for Polish structure – for MD = 20%, Ph = 10% and Pz = 20%.

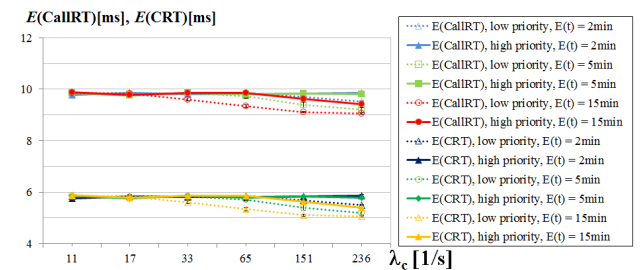


FIGURE 7. Mean Call Release Time and Connection Release Time for Polish structure – for MD = 20%, Ph = 10%, Pz = 20%.

The basic analyze in forms of displaying vector and scalar data and histograms is also provided by the OMNeT++ Integrated Development Environment (IDE).

The mean Call Set-up Time  $E(CallST)$ , mean Connection Set-up Time  $E(CST)$  for Polish structure are presented in Fig. 6. The mean Call Set-up Time  $E(CallST)$  for Polish structure does not exceed 15 milliseconds while the mean Connection Set-up Time  $E(CST)$  does not exceed 11 milliseconds. The mean values of call/connections times for the same  $E(t)$  are smaller for low priority requests compared to high priority requests. For requests intensity greater than 33 requests per second  $E(CallST)$  and  $E(CST)$  are slightly smaller for low priority requests than for high priority requests. Detailed analysis of the length of connection paths indicated that for low priority requests connections paths are shorter than for high ones for values of requests intensities higher than 33 requests per second.

In Fig. 7 mean Call Release Time  $E(CallRT)$  and mean Connection Release Time  $E(CRT)$  for Polish structure



are presented. The mean Call Release Time  $E(\text{CallRT})$  for Polish structure does not exceed 10 milliseconds while the mean Connection Release Time  $E(\text{CRT})$  does not exceed 6 milliseconds.

In our research we take into consideration the offered traffic to the transport plane ( $A_0$ ) and the offered traffic per inter-domain link ( $a_0$ ) which are calculated as follows:

$$A_0 = \lambda_c \cdot E(t) \cdot N_R \quad (1)$$

$$a_0 = \frac{\lambda_c \cdot E(t) \cdot N_R \cdot C_{MD}}{N_{BU}} \quad (2)$$

where:

- $\lambda_c$  - total intensity request per second,
- $E(t)$  - mean connection duration time,
- $N_R$  - average number of bandwidth units (BU) per connection request,
- $C_{MD}$  - coefficient of inter-domain traffic in the all offered traffic,
- $N_{BU}$  - the number of bandwidth units in the inter-domain link.

Our research indicated that for higher intensities we obtained higher loss probabilities for low priority requests. Comparing  $E(\text{CallST})$  and  $E(\text{CST})$  for  $E(t) = 2$  minutes versus loss probabilities for  $Ph = 10\%$  and  $Pz = 20\%$  (Fig. 8) we noticed that loss probabilities for low priority requests differ significantly from high priority requests. For offered traffic to the transport plane  $A_0 = 56679$  Erl and  $Ph = 10\%$  loss probability for high priority requests equals 0.0027 while for low priority requests equals 0.24. For  $E(t) = 5$  minutes,  $Ph = 10\%$  and  $Pz = 20\%$  for offered traffic to the transport plane  $A_0 = 90348$  Erl loss probability for high priority requests equals 0.0087 Erl while for low priority requests equals 0.45.

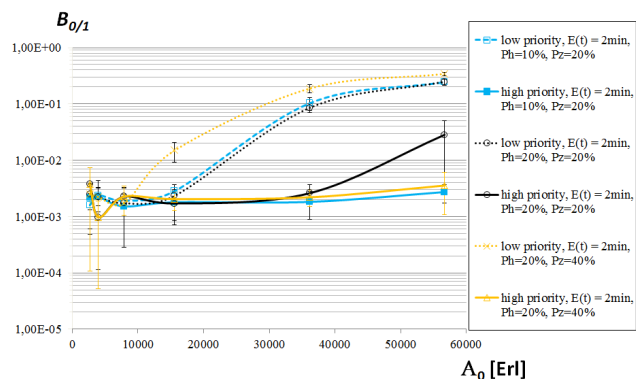


FIGURE 8. Loss probability for low (0) and high (1) priority requests for  $E(t) = 2$  minutes for Polish structure.

As depicted in Fig. 8 and Fig. 9 for the same resource allocation ( $Pz = 20\%$ ) loss probability for  $Ph = 20\%$  does not differ significantly from results obtained for  $Ph = 10\%$  for low priority requests. Confidence intervals of loss probabilities for low priority requests overlap as well as for  $E(t)$

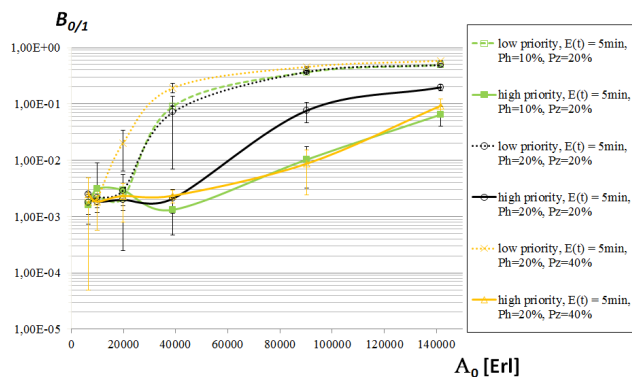


FIGURE 9. Loss probability for low (0) and high (1) priority requests for  $E(t) = 5$  minutes for Polish structure.

equals 2 minutes and  $E(t)$  equals 5 minutes. High priority requests probability differences are significant and  $B_1$  equals 0.076 for  $A_0 = 90348$  Erl,  $Pz = 20\%$ ,  $Ph = 20\%$  while for  $A_0 = 90348$  Erl,  $Pz = 20\%$ ,  $Ph = 10\%$ ,  $B_1$  equals 0.0087. As expected, the change of resource allocation for high priority requests from  $Pz = 20\%$  to  $Pz = 40\%$  decreases loss probabilities for high priority requests and provides to increase probabilities for low priority requests.

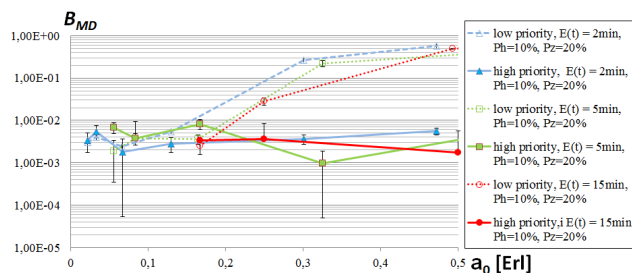


FIGURE 10. Loss probability  $B_{MD}$  as a function of offered traffic per inter-domain link for  $Ph = 10\%$ ,  $Pz = 20\%$ , and  $MD = 20\%$  for Polish structure.

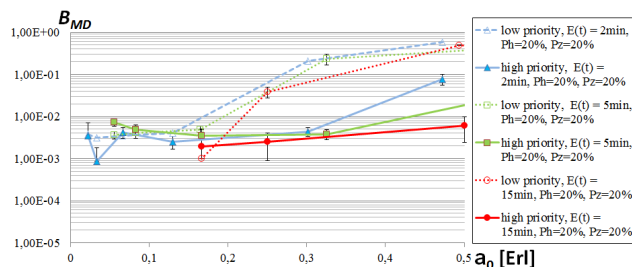


FIGURE 11. Loss probability  $B_{MD}$  as a function of offered traffic per inter-domain link for  $Ph = 20\%$ ,  $Pz = 20\%$ , and  $MD = 20\%$  for Polish structure.

Based on simulation results loss probabilities for inter-domain connections ( $B_{MD}$ ) with a distinction between high and low priority requests are determined and presented in Fig. 10 and Fig. 11 for inter-domain traffic 20% and  $Ph = 10\%$  and  $Ph = 20\%$ . In condition of  $Ph = 10\%$  loss probabilities for high priority requests are smaller than for  $Ph = 20\%$  for the same value of offered traffic per inter-domain link.

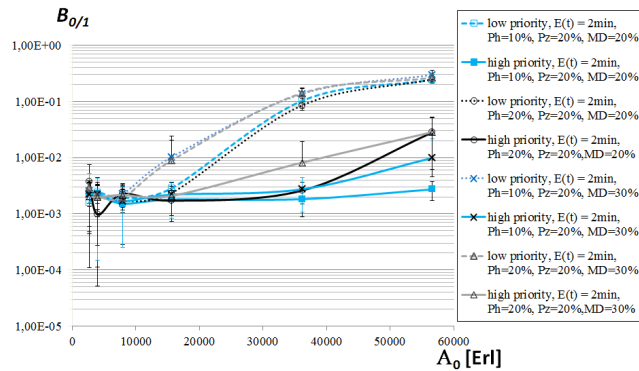


FIGURE 12. Loss probability as a function of offered traffic for MD = 20% and 30% for Polish structure.

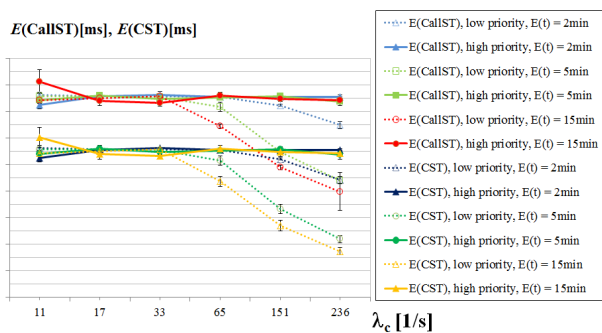


FIGURE 13. Mean Call Set-up Time and Connection Set-up Time for European structure – for MD = 20%, Ph = 10% and Pz = 20%.

Performed research for Polish structure for MD = 20% and MD = 30% are presented in Fig. 12. In this case MD has impact on loss probability for low and high priority requests and leads to loss probability increase for low and high priority.

Taking under consideration the European structure we can notice that the mean Call Set-up Time  $E(\text{CallST})$  does not exceed 35.5 milliseconds while the mean Connection Set-up Time  $E(\text{CST})$  does not exceed 31 milliseconds (Fig. 13). The mean values of call/connections times for the same  $E(t)$  are smaller for low priority requests compared to high priority requests.

For requests intensity higher than 33 requests per second  $E(\text{CallST})$  and  $E(\text{CST})$  are slightly smaller for low priority requests than for high priority requests for the same reason as for Polish structure.

In Fig. 14 mean Call Release Time  $E(\text{CallRT})$  and mean Connection Release Time  $E(\text{CRT})$  for European structure are presented. The mean Call Release Time  $E(\text{CallST})$  for European structure does not exceed 28 milliseconds while the mean Connection Release Time  $E(\text{CST})$  does not exceed 23 milliseconds.

For offered traffic to the transport plane  $A_0 = 56679$  Erl and  $Ph = 10\%$  and  $Pz = 20\%$  loss probability for high priority requests equals 0.0025 while for low priority requests equals 0.15. For  $E(t) = 5$  minutes,  $Ph = 10\%$  for offered traffic to the transport plane  $A_0 = 90348$  Erl loss probability for high

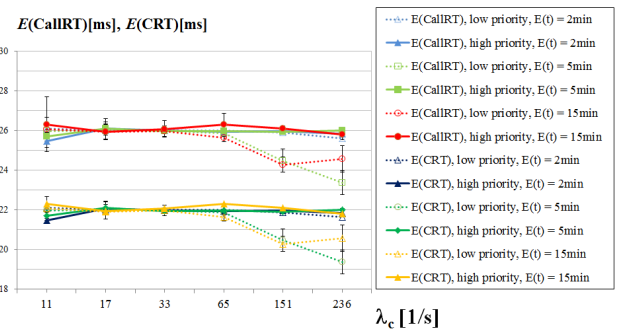


FIGURE 14. Mean Call Release Time and Connection Release Time for European structure – for MD = 20%, Ph = 10%, Pz = 20%.

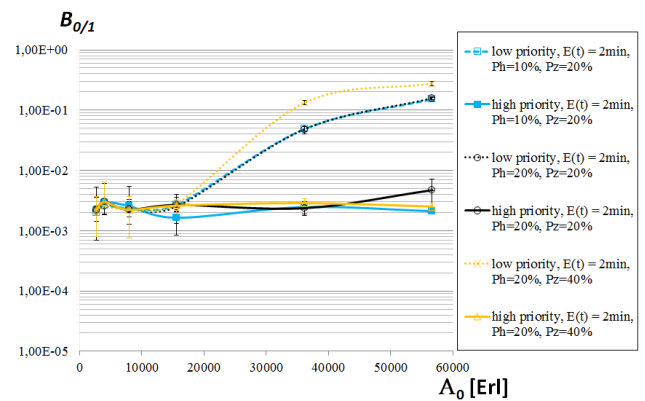


FIGURE 15. Loss probability for low (0) and high (1) priority requests for  $E(t) = 2$  minutes for European structure.

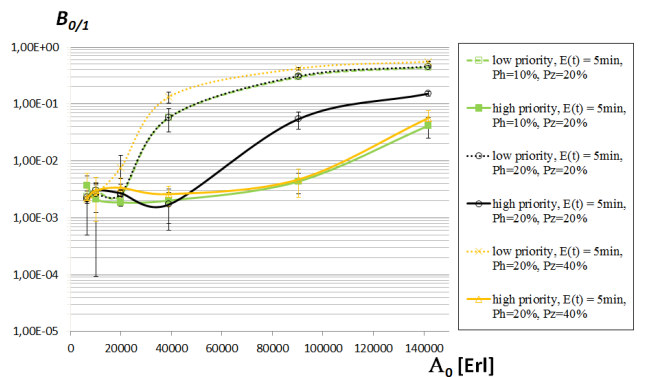


FIGURE 16. Loss probability for low (0) and high (1) priority requests for  $E(t) = 5$  minutes for European structure.

priority requests equals 0.0047 while for low priority requests equals 0.3.

As depicted in Fig. 15 and Fig. 16. for the same resource allocation ( $Pz = 20\%$ ) loss probabilities for  $Ph = 20\%$  do not differ significantly from results obtain for  $Ph = 10\%$  for low priority requests. Confidence intervals of loss probabilities for low priority requests overlap as well as for  $E(t)$  equals 2 minutes and  $E(t)$  equals 5 minutes.

Loss probability presented in Fig. 15 and Fig. 16 for low priority requests differs significantly from high priority requests.

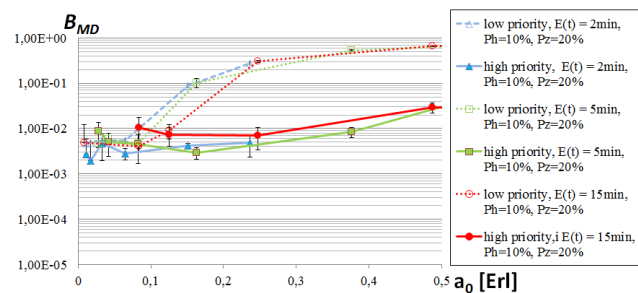


FIGURE 17. Loss probability  $B_{MD}$  as a function of offered traffic per inter-domain link for  $Ph = 10\%$ ,  $Pz = 20\%$ , and  $MD = 20\%$  for European structure.

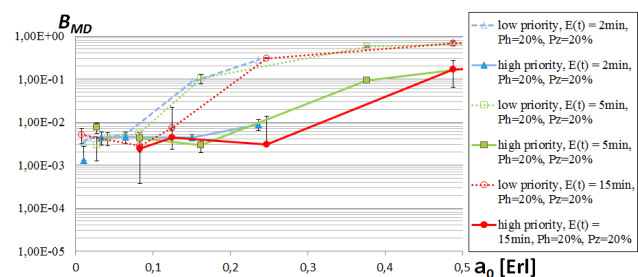


FIGURE 18. Loss probability  $B_{MD}$  as a function of offered traffic per inter-domain link for  $Ph = 20\%$ ,  $Pz = 20\%$ , and  $MD = 20\%$  for European structure.

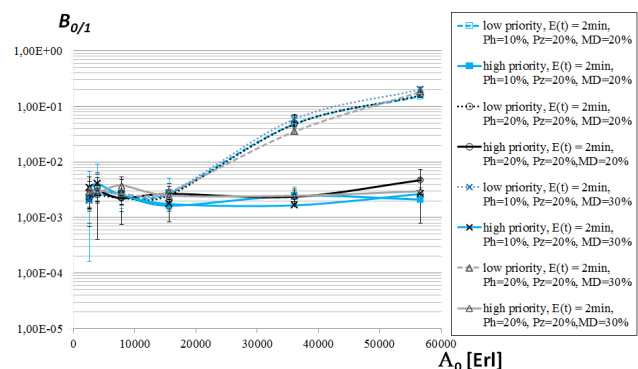


FIGURE 19. Loss probability as a function of offered traffic for  $MD = 20\%$  and  $30\%$  for European structure.

The change of resource allocation for low priority requests from  $Pz = 20\%$  to  $Pz = 40\%$  significantly affects loss probabilities for high priority requests while loss probabilities for low priority requests are higher, so the tendency is the same as in case of Polish structure. Loss probabilities for European structure for inter-domain connections ( $B_{MD}$ ) with a distinction between high and low priority requests are determined and presented in Fig. 17 and Fig. 18 for inter-domain traffic  $20\%$ ,  $Ph = 10\%$  and  $Ph = 20\%$ .

In condition of  $Ph = 10\%$  for offered traffic smaller than  $0.15$  Erl loss probability for low priority requests does not exceed  $0.1$ . In compare with Polish structure loss probability lower than  $0.1$  is obtained up to  $0.25$  Erl offered traffic per inter-domain link.

Performed research for European structure for  $MD = 20\%$  and  $MD = 30\%$  are presented in Fig. 19.

Loss probability for  $MD = 30\%$  does not differ significantly for loss probabilities for  $MD = 20\%$ . Confidence intervals of loss probabilities for  $MD = 20\%$  and  $MD = 30\%$  overlap for high priority requests as well as for low priority requests.

The presented results for Polish and European structures are in accordance with expectations and can be logically explained.  $E(\text{CallST})$ ,  $E(\text{CST})$  for intensities higher than  $33$  requests per second are longer for high priority requests than for low priority requests. Due to the fact that pool approach was implemented in resource reservation algorithm. Implemented resource algorithm gives more capabilities to supply wavelength continuity constraint condition and loss probability for high priority requests is smaller.

## V. CONCLUSION

In the paper we proposed the simulator for performance evaluation of ASON/GMPLS network. The descriptions of functional blocks of ASON/GMPLS simulator and implementation details were presented. We described examples of scenarios and simulations results for Polish and European structures to indicate the complex interrelations in performance evaluation of ASON/GMPLS hierarchical control plane. The variety of results was provided to illustrate simulator capabilities and its usability in ASON/GMPLS network examination. The simulation results were presented to show the versatility and the possibility of easy adjustment to various simulations conditions.

The simulator is suitable for modeling and simulating different ASON/GMPLS network structures with the variety of functionalities which can be easily extended according to the needs. The network operator and researcher can use the simulator in order to evaluate ASON/GMPLS network performance and capabilities. Moreover, the modules of ASON/GMPLS simulator are implemented with the C++ programming language in an open source OMNeT++ environment. This increases the flexibility in extension of the simulator functionality for Next Generation Networks and SDN architecture.

The presented simulator is currently expanding in order to convergent SDN and ASON/GMPLS architectures. ASON/GMPLS and SDN functionality are aimed in the same direction. Many features provided by ASON/GMPLS could be run for cooperation with SDN controller.

## REFERENCES

- [1] *Architecture for the Automatically Switched Optical Network (ASON)*, document ITU-T Rec. G.8080/Y.1304, Feb. 2012.
- [2] A. Farrel and I. Bryskin, *GMPLS, Architecture and Applications*. Berlin, Germany: Morgan Kaufmann, 2006.
- [3] *Generalized Multi-Protocol Label Switching (GMPLS) Architecture*, document IETF RFC 3945, E. Mannie, Oct. 2004.
- [4] *Common Control Aspects. Amendment 1*, document ITU-T Rec. G.7701 Amendment 1, Mar. 2018.
- [5] *Architecture for SDN Control of Transport Networks*, document ITU-T Rec. G.7702, Mar. 2018.
- [6] C. Zhang, J. Li, H. Wang, A. Guo, and C. Janz, "Evaluation of dynamic optical service restoration on a large-scale ROADM mesh network," *IEEE Commun. Mag.*, vol. 57, no. 4, pp. 138–143, Apr. 2019, doi: 10.1109/MCOM.2019.1800307.

- [7] S. Rumley, C. Gaumier, R. Aparicio-Pardo, C.-H. Chang, W. Colitti, B. García-Manrubia, P. Kourtessis, J. A. Martínez-León, A. Nowè, P. Pavón-Mariño, J. Scharf, and K. Steenhaut, "Simulation of ASON/GMPLS using OMNET++ simulator," in *Towards Digital Optical Networks: COST Action 291 Final Report*. Berlin, Germany: Springer, 2009, pp. 344–349.
- [8] *Distributed Call and Connection Management: Signalling Mechanism Using GMPLS RSVP-TE*, document ITU-T Rec. G.7713.2/Y.1704.2, Mar. 2003.
- [9] V. Fajardo, J. Arkko, J. Loughney, and G. Zorn, *Diameter Base Protocol*, document IETF RFC 6733, Oct. 2012.
- [10] D. Li, D. Ceccarelli, and L. Berger, *Link Management Protocol Behavior Negotiation and Configuration Modifications*, document IETF RFC 6898, Mar. 2013.
- [11] *ASON Routing Architecture and Requirements for Link State Protocols*, document ITU-T Rec. G.7715.1/Y.1706.1, Feb. 2004.
- [12] S. Kaczmarek and M. Młynarczuk, "The performance of ASON/GMPLS network with hierarchical control plane structure," in *Proc. Int. Conf. Softw., Telecommun. Comput. Netw. (SoftCOM)*, Sep. 2019, pp. 19–21, doi: [10.23919/SOFTCOM.2019.8903595](https://doi.org/10.23919/SOFTCOM.2019.8903595).
- [13] S. Kaczmarek and M. Młynarczuk, "Quality of service in ASON/GMPLS network with hierarchical control plane structure," in *Proc. Int. Conf. Softw., Telecommun. Comput. Netw. (SoftCOM)*, Sep. 2020, pp. 17–19, doi: [10.23919/SoftCOM50211.2020.9238242](https://doi.org/10.23919/SoftCOM50211.2020.9238242).
- [14] M. Rupp, S. Schwarz, and M. Taranez, "The Vienna LTE-advanced simulators: Up and downlink, link and system level simulation," in *Signals and Communication Technology*. Singapore: Springer, 2016, doi: [10.1007/978-981-10-0617-3](https://doi.org/10.1007/978-981-10-0617-3).
- [15] M. M. Nasralla, I. García-Magariño, and J. Lloret, "MASEMUL: A simulation tool for movement-aware MANET scheduling strategies for multimedia communications," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–12, Mar. 2021, doi: [10.1155/2021/6651402](https://doi.org/10.1155/2021/6651402).
- [16] G. E. Kylafas and J. K. Soldatos, "Asons: An automatically switched optical networks simulator," *Comput. Commun.*, vol. 28, no. 2, pp. 224–235, Feb. 2005, doi: [10.1016/j.comcom.2004.09.015](https://doi.org/10.1016/j.comcom.2004.09.015).
- [17] D. Papadimitriou, J. Drake, J. Ash, A. Farrel, and L. Ong, *Requirements for Generalized MPLS (GMPLS) Signaling Usage and Extensions for Automatically Switched Optical Network (ASON)*, document IETF RFC 4139, Jul. 2005.
- [18] *Signaling Protocol Interworking of ASON/GMPLS Network Domains*, document OIF Guideline, Jun. 2008.
- [19] A. Malis, A. Lindem, and D. Papadimitriou, *Automatically Switched Optical Network (ASON) Routing for OSPFv2 Protocols*, document IETF RFC 6827, Jan. 2013.
- [20] P. Luo, S. Huang, W. Lian, B. Guo, and W. Gu, "Efficient routing algorithms for hierarchical ASON," in *Proc. 35th Euro. Conf. Opt. Commun. (ECOC)*, Vienna, Austria, 2009, pp. 1–2.
- [21] O. S. Yeremenko, O. V. Lemeshko, O. S. Nevzorova, and A. M. Hailan, "Method of hierarchical QoS routing based on network resource reservation," in *Proc. IEEE 1st Ukraine Conf. Electr. Comput. Eng. (UKRCON)*, Kiev, Ukraine, May 2017, doi: [10.1109/UKRCON.2017.8100393](https://doi.org/10.1109/UKRCON.2017.8100393).
- [22] S. Araki and I. Nishioka, "Multi-domain ASON/GMPLS network operations," in *Proc. Int. Conf. Photon. Switching*, Sapporo, Japan, Aug. 2008, pp. 4–7, doi: [10.1109/PS.2008.4804268](https://doi.org/10.1109/PS.2008.4804268).
- [23] R. Alvizu, G. Maier, N. Kukreja, A. Pattavina, R. Morro, A. Capello, and C. Cavazzoni, "Comprehensive survey on T-SDN: Software-defined networking for transport networks," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2232–2283, 4th Quart. 2017, doi: [10.1109/COMST.2017.2715220](https://doi.org/10.1109/COMST.2017.2715220).
- [24] Y. Li and D. C. Kilper, "Optical physical layer SDN [Invited]," *J. Opt. Commun. Netw.*, vol. 10, no. 1, pp. A110–A121, Jan. 2018, doi: [10.1364/JOCN.10.00A110](https://doi.org/10.1364/JOCN.10.00A110).
- [25] S. Kaczmarek, M. Narloch, M. Młynarczuk, and M. Sac, "The realization of NGN architecture for ASON/GMPLS network," *J. Telecomm. Inf. Technol.*, vol. 2011, no. 3, pp. 47–56, Mar. 2011.
- [26] *OMNet++ Network Simulation Framework*. Accessed: Jul. 8, 2021. [Online]. Available: <https://omnetpp.org>
- [27] *OMNet++ Simulation Manual*. Accessed: Jul. 8, 2021. [Online]. Available: <https://doc.omnetpp.org/omnetpp/manual>
- [28] S. Orłowski, R. Wessälly, M. Pióro, and A. Tomaszewski, "SNDlib 1.0-survivable network design library," *Netw., Int. J.*, vol. 55, no. 3, pp. 276–286, May 2010.
- [29] A. Szymanski, A. Lason, J. Rzasa, and A. Jajszczyk, "Grade-of-service-based routing in optical networks [quality-of-service-based routing algorithms for heterogeneous networks]," *IEEE Commun. Mag.*, vol. 45, no. 2, pp. 82–87, Feb. 2007, doi: [10.1109/MCOM.2007.313400](https://doi.org/10.1109/MCOM.2007.313400).



**SYLWESTER KACZMAREK** received the M.Sc. degree (Hons.) in electronics engineering and the Ph.D. (Hons.) and D.Sc. degrees in switching and teletraffic science from Gdansk University of Technology (GUT), Poland, in 1972, 1981, and 1994, respectively.

From 1972 to 1981, he worked as an Assistant with the Faculty of Electronics, Telecommunications, and Informatics (FETI), GUT, where he held the position of an Assistant Professor, from 1981 to 1994, and an Associate Professor, from 1994 to 2005. Since 2005, he has been working as a Professor with FETI. From 2007 to 2018, he was the Head of the Teleinformation Networks Department, FETI. So far, he has published two monographs and more than 250 articles. Among his promoted students, there are 278 engineers and M.Sc.s., and four Ph.D. students. He also participates and directs projects and research-development grants. His research interests include IP QoS, GMPLS, ASON and SDN networks, switching, QoS routing, teletraffic, and multimedia services. His research is currently focused on developing and applicability of VoIP, IMS/NGN, and SDN technologies. Among his distinctions, he is also a TPC member of conferences, such as ITNAC, ISCSIS, and KSTiT.



**MAGDALENA MŁYNARCZUK** received the M.Sc.Eng. degree in telecommunication systems and networks from the Faculty of Electronics, Telecommunications, and Informatics (FETI), Gdansk University of Technology (GUT), in 2004, and the M.Sc. degree in economy and finance from the Faculty of Management and Economics, GUT, in 2007.

From 2008 to 2011, she was an Assistant with FETI, GUT, where she was also a Lecturer with the Department of Teleinformation Networks, from March 2011 to September 2017. Since October 2017, she has been working as an Assistant with FETI, GUT, where she gives classes related to her research areas and finishes her Ph.D. thesis titled "Routing QoS in multi-domain optical network with hierarchical structure control plane." She has also actively participated in several research and development projects/grants conducted with the Department of Teleinformation Network. Her research interests include control of optical networks, transmission and switching technology, teletraffic engineering, and voice over IP technology.

• • •