

VĚDECKÉ SPISY VYSOKÉHO UČENÍ TECHNICKÉHO V BRNĚ

Edice PhD Thesis, sv. 619

ISSN 1213-4198

thesis IS

Ing. Jan Beran

**Performance Analysis
in Ip-Based Industrial
Communication Networks**

BRNO UNIVERSITY OF TECHNOLOGY
Faculty of Electrical Engineering and Communication
Department of Control and Instrumentation

Ing. Jan Beran

PERFORMANCE ANALYSIS IN IP-BASED INDUSTRIAL
COMMUNICATION NETWORKS

ANALÝZA VÝKONNOSTI V IP PRŮMYSLOVÝCH
KOMUNIKAČNÍCH SÍTÍCH

AUTOREFERÁT DIZERTAČNÍ PRÁCE

Obor: Kybernetika, automatizace a měření
Školitel: prof. Ing. František Zezulka, CSc.
Oponenti: prof. Ing. Miroslav Švéda, CSc., YUT v Brně
doc. Dr. Ing. Zdeněk Hanzálek, CVUT v Praze
Datum obhajoby: 20. prosince 2010, Brno

Klíčová slova:

výkonnostní analýza, průmyslová komunikace, network calculus, kvalita služeb, IP směrovač, TestQoS, virtuální automatizační sítě

Keywords:

performance analysis, industrial communication, network calculus, quality of service, IP router, TestQoS, Virtual Automation Networks

Místo uložení:

Vědecké oddělení děkanátu FEKT VUT v Brně, Údolní 53, Brno, 602 00

© Jan Beran, 2011

ISBN 978-80-214-4268-9

ISSN 1213-4198

Contents

1	INTRODUCTION	5
1.1	Main Objectives	5
1.2	Related Work	5
2	STATE OF THE ART	6
2.1	Internetworking Technologies	6
2.2	Communication in Industrial Automation	6
2.3	Networking Device Architectures	6
2.3.1	Switches	6
2.3.2	IP Routers	7
2.4	Quality of Service	7
2.5	Network Calculus	8
3	EMPIRICAL ANALYSIS OF PERFORMANCE BOUNDS	11
3.1	TestQoS: Quality of Service Test Bed	11
3.2	Measurement Results	11
4	NETWORKING DEVICE MODELLING	13
4.1	Rate-Variable-Latency Service Curve	13
4.2	Definition of Model Structure	13
4.2.1	Switch Fabric Model Structure	14
4.2.2	Outgoing Interface Model Structure	15
4.3	Port-to-Port Service Curve of a Networking Device	16
4.3.1	PBOO/PMOO Approach	16
4.3.2	Extended PBOO Approach	17
4.4	Identification of the Model Parameters	18
4.4.1	Switch-Fabric Parameters	18
4.4.2	Outgoing Interface Parameters	19
5	VALIDATION OF MODELS OF NETWORKING DEVICES	20
5.1	HP ProCurve 1800-8G Switch	20
5.1.1	Switch Parametrisation	20
5.1.2	Port-To-Port Service Curve	20
5.1.3	Model Simulation	21
5.2	Cisco 2811 Router with HWIC-2FE Module	22
5.2.1	Switch Fabric Parametrisation	23
5.2.2	Outgoing Interface Parametrisation	23
5.2.3	Port-to-Port Service Curves	24
5.2.4	Model Simulation	25
6	ASSESSMENT AND CONCLUSIONS	27
6.1	Validation Assessment	27
6.2	Conclusions	27
6.3	Unresolved Issues and Further Research	28
6.4	Final Remarks	28

1 INTRODUCTION

With the growing scale of control systems and their distributed nature, communication networks have been gaining importance and new research challenges have been appearing. Zampieri categories in [34] the upcoming research challenges in terms of networked control systems (NCS), i.e., feedback control systems using packet-oriented networks, such as Ethernet. The major problem, contrary to previously used control systems with dedicated communication circuits, is the time-varying delay of control and measurement signals. Hence, the major efforts according to Zampieri are: control of networks (i) and control over networks (ii). There are many efforts within the latter domain trying to design the control systems in a way to decrease susceptibility of control to signal latency. For instance, Chow proposes in [25] an optimal regulator taking the networking aspects as a design criterium.

On the other hand, the former effort refers to the real-time aspects of the communication networks and their control by means of quality of service (QoS) mechanisms providing on-demand resource sharing of the networking devices. In a local scope, NCS in industrial environment are most often based on Ethernet-based fieldbuses. Producers of fieldbuses have managed to handle real-time communication aspects and deliver the expected behaviour in a scalable way. Summary of the existing solutions can be found in [13]; in larger communication scopes, the reliable industrial solutions emerge only slowly despite customers' requirements [2].

Virtual Automation Networks (VAN) project introduced in [3] was an integrated project within the 6th Framework Programme dealing with future communication technologies in industrial automation. One of the key topics in the project was specification of a real-time framework for network topologies with enterprise-wide communication scope. The approach adopted in the project was employment of IP-based infrastructures with QoS capabilities used in telecommunication networks.

IP-based communication architectures are not very prone to adoptions which would provide a perfect match with the industrial requirements on real-time behaviour. Hence, it is the opinion of the author that a modelling framework assisting performance analysis of the temporal behaviour would leverage the existing commercial-off-the-shelf (COTS) networking devices, such as routers and switches, and would encourage their use in industrial automation.

1.1 Main Objectives

Hence, the main objective of the dissertation is *establishment of a relevant modelling framework based on network calculus which will assist worst-case performance analysis of temporal behaviour of IP-based industrial communication networks.*

The high-level objective was broken down to partial milestones necessary to be met in order that the high-level objective is fulfilled:

Empirical Analysis is used for several reasons. Firstly, it is necessary to identify dominant factors influencing the temporal performance of networking devices for model parameter identification. Secondly, they are used for final model validation.

Predefined Model Structures are to be used for rapid model structure design of networking devices. Based on the investigations of networking device architectures introduced in Section 2.3, basic model structures are to be developed and port-to-port service quantifications inferred.

Methods for Parameter Identification have to be defined and applied as a solution to this topic has not been tackled by researchers so far. This objective is well aligned with identifications of the upcoming research directions recognised by Schmitt in [31].

Device Concatenation strategy for concatenation of networking devices in order to perform complex networks' analysis.

1.2 Related Work

The main inspiration has been taken from Georges who made similar attempt to model switched infrastructure in [19], [20], [21], [22], and [23]. Contrary to the proposed work, his measurements were less precise and the models were not based on min-plus algebra. Similarly, Jasperneite has provided classification of different types of industrial flows and application of network calculus on industrial Ethernet applications in [27] and [28]. Entirely white-box modelling of a network processor using network calculus has been proposed in [16]. Profiling of the model structure and parameter identification based on measurement of networking devices is outlined in [12].

2 STATE OF THE ART

2.1 Internetworking Technologies

Internetwork is a network shielding several individual networks and interconnects them with intermediate networking devices [18]. The individual networks can be based on different standardised technologies. *Internetworking* refers to the effort, products, and procedures that meet the challenge of creating and administering internetworks [18]. The main challenges of internetworking are connectivity, reliability, security, and network management. Internetworks can be coarsely divided into two groups. *Local Area Networks (LANs)* enable multiple users in a close geographical area to access information and share common resources. Nowadays, LANs are predominantly based on the Ethernet technology. *Wide Area Networks (WANs)* interconnect spatially dispersed LANs. WANs are tailored to higher traffic aggregation and rely on less shared resources. WANs are usually infrastructures provided by internet service providers (ISP). One can recognise *access networks* and *core networks*. Run-time industrial communication is, apart from exceptions, related to LANs. Indeed, there are special needs of industrial automation to employ WAN technologies. However, real-time expectations are not too high. On the other hand, LANs are subject to performance analyses and correspond to the scope in which real-time communication is essential. Industrial automation has usually made use of physical and link (e.g. Ethernet), and application layer due to reduced temporal overhead, and intrinsically simpler topology than is usual with internetworks. Ethernet-based fieldbuses follow this approach in their soft-real-time modes.

2.2 Communication in Industrial Automation

Industrial automation is based mainly on *fieldbuses*. Fieldbus is the lowest level industrial network in computer communication hierarchy of factory automation and process control systems [26]. Apart from the communication mission, a fieldbus also contains a communication object model (COM) which provides the engineer with a certain level of abstraction. Legacy fieldbuses used are based on proprietary communication technologies or aging standards, such as RS485.

With the growth of Ethernet the legacy fieldbuses regarded in [29] are being slowly yet steadily hindered by the upcoming versions - *Ethernet-based fieldbuses*, which are becoming a state of the art. A self-documenting evidence is the existence of standardised solutions as introduced in IEC 61158 Type 10 to Type 16 (Profinet IO, EtherCAT, Ethernet Powerlink, EPA, and SERCOS III, respectively). Ethernet has been adopted for its high availability, low cost, and compatibility with office communications. Ethernet makes integration possible across all levels of industrial automation, reaching from Enterprise Resource Planning (ERP) over Manufacturing Execution System (MES) down to operator level and process instrumentation. Nowadays fieldbuses are very complex systems providing an application engineer with a vast variety of tools and functionalities.

2.3 Networking Device Architectures

Internetworking technologies employ a plenty of networking device types providing data transmission between end devices. Industrial automation has adopted only a subset of the networking devices, and occasionally, provided adaptations. Network devices which are used in the state-of-the-art industrial automation and network devices considered for future applications are introduced in this section. The investigation is focused on the aspects related to the real-time behaviour as these are important for further performance analysis and modelling of temporal behaviour. Basic findings in this field are summarised in [4]. Moreover, only IP-based Ethernet devices are considered.

2.3.1 Switches

Switch is a network device operating at link layer (L2). The extension to hub is that a switch accommodates a forwarding table caching the physical addresses of the ambient network devices paired with the a number of the network interfaces to which the devices are connected. Ethernet switch functionality is standardised by IEEE 802.1 standard family. From the QoS point of view, two standard extensions are to mention. 802.1q extension defines Virtual Local Area Network (VLAN). Furthermore, 802.1p standard defines seven classes of traffic corresponding the VLAN priorities and thus giving them real semantics.

Figure 1 depicts a generalised architecture of a four-port Ethernet switch. Incoming interfaces (II) pass frames to a packet forwarding unit (PFU). PFU resolves the outgoing port by consulting a forwarding table. Consequently, the frame can be forwarded through a switch fabric (SF) to a specific outgoing interface (OI) and finally to the interface hardware.

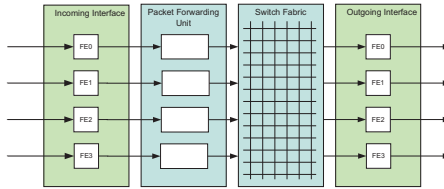


Figure 1: General switch architecture

2.3.2 IP Routers

Router is a networking device providing interconnection of network segments based network layer (L3) information, in our case IP. Routers can accommodate an immense amount of features and protocols. However, there are two mandatory tasks a router has to provide: *routing* and *forwarding* [11].

Routing refers to determining the optimal path a packet should take to reach the destination end device. Routers exchange routing information among each other to collaboratively reach the optimal path based on the routing metrics. The result is stored in a routing table of a router. Details can be found in [18], [15], and [11]. Routing is not a time-critical task.

Forwarding is a process of transferring a packet from an incoming interface to an outgoing interface based on the records in Routing Table (RT). Forwarding is time-critical and is decisive for the total throughput of traffic through the switch.

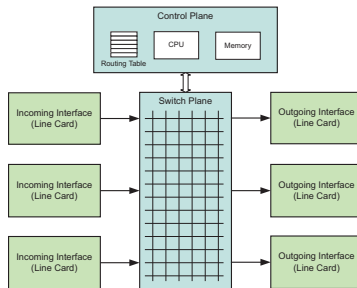


Figure 2: General router architecture

A general router architecture is shown in Figure 2. However, specific architectures differ. A router contains physical incoming and outgoing interfaces (IIs and OIs). The interfaces range from the simplest similar to those in office PCs to very sophisticated housing forwarding modules, queue management features, various hardware accelerators, etc. Packet forwarding from II to OI is provided by a switch fabric (SF). SF can be implemented in various ways (shared-memory, crossbar, etc.). Implementation of switch plane is decisive for managing flow aggregation and traffic congestions. Finally, control plane takes care of configuration, routing, management of routing table and providing the routing information to forwarding modules. Details can be found in the dissertation.

2.4 Quality of Service

QoS refers to the capability of a network to provide better service to selected network traffic over various technologies [18]. The primary goal of QoS is to provide priority including dedicated bandwidth, controlled jitter and latency [18]. A common metrics is defined to compare the QoS level delivered by the QoS mechanisms. The mostly used quadruple of metrics for the QoS evaluations is bandwidth, packet delay, jitter, and packet loss. There are two general mechanisms providing a required QoS level: Integrated Services (IntServ) and Differentiated Services (DiffServ).

DiffServ is a class-based mechanism. Its priority treatment is coarser than IntServ's as it works with classes rather than flows. The information of the class is located in the Differentiated Service Code Point (DSCP) in

the IP packet header. The architecture was established by [9] and updated by [24]. DiffServ reflects better trends in Internetworking QoS. Therefore, it is employed for the proposed application in industrial automation. See [5].

DiffServ mechanisms are employed in a *DiffServ domain*. It uses four basic types of mechanisms: *Classification*, *Traffic Shaping*, *Congestion Management*, and *Congestion Avoidance*. The following two were considered in the proposed work:

Classification is a process of labelling the incoming packet with a proper DSCP code. Classification consists in two steps: *identification* and *marking*. Classification is performed on a DiffServ ingress, typically at a L3-switch or an edge router. An emerging trend is to classify the traffic directly at the traffic source, i.e., at a connected device itself. The coding scheme has to be provided by user. The DSCP mapping is designed with regard to backward compatibility with its predecessor *ToS* and *IP precedence* fields defined in [1].

Congestion management (CM) is the core of the DiffServ architecture. Firstly, *queuing* handles buffering of packets to appropriate queues is based on the DSCP code in case of congestion. *Packet scheduling* represent polling of packets from the queue and passing them either directly to the OI or output buffer. Choosing a proper packet-scheduling algorithm is a trade-off between efficiency, fairness, determinism, and starving of low-priority traffic. The main principles of packet scheduling are first-in-first-out (FIFO), priority queueing (PQ), round-robin mechanisms such as custom queueing (CQ) and weighted-round-robin (WRR), or finally generalised-processor-sharing (GPS) based mechanisms, such as weighted-fair queueing (WFQ).

2.5 Network Calculus

Network calculus is a theory of deterministic queuing systems found in computer networks. A comprehensive summary on evolution of network calculus can be found in [17]. In short, network calculus can be used as a block algebra where the signals represent the data flows and the systems represent the services offered to the flows.

A flow can be characterised by a rate function $r(t)$ representing a number of bits per second passing through a point of observation at a moment t . In network calculus, flows are characterised by a *cumulative rate function* $R(t)$ which is defined as $R(t) = \int_0^t r(\tau) d\tau$. As network calculus deals with worst-case performance analysis, one works with upper bounds of the flow and system characteristics. With flow characteristics, one defines an *arrival curve* which upper-bounds the cumulative rate-function by a definition introduced in [10, 8]:

Definition 2.1 (Arrival Curve) *Given a wide-sense increasing function α defined for $t \geq 0$ (namely, $\alpha \in \mathcal{F}$), we say that the flow R is constrained by α if and only if for all $s \leq t$: $R(t) - R(s) \leq \alpha(t - s)$. We say that R has α as an arrival curve, or also that R is α -smooth.*

The most often used arrival curve is the *affine function* $\gamma_{r,b}$ defined as

$$\gamma_{r,b}(t) = \begin{cases} rt + b & \text{if } t > 0 \\ 0 & \text{otherwise,} \end{cases}$$

introduced in [10, 128]. The r parameter represents the average rate of the flow throughout the period of observation. The b parameter represents backlog (virtually pending traffic) of the flow.

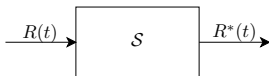


Figure 3: Service element

Flow characteristics may change after having passed a service element. Service element is an atomic input-output block depicted in Figure 3 described by a service curve. One can describe more complex system using these service elements. Definition 2.2 introduced in [10, 23] defines a service curve of a service element.

Definition 2.2 (Service Curve) *Consider a system S and a flow through S with input and output function R and R^* . We say that S offers to the flow a service curve β (if and only if $\beta \in \mathcal{F}$) and $R^*(t) \geq (R \otimes \beta)(t)$.*

The most often used service element is the rate-latency service curve defined as

$$\beta(t) = \beta_{R,T} = R[t - T]^+ = \begin{cases} R(t - T) & \text{if } t > T \\ 0 & \text{otherwise} \end{cases}$$

The R parameter represent the minimum rate with which the element is able to process traffic and the T parameter represents the maximum latency of a bit passing through the element. Further service curves exist: $\lambda_R(t) = \beta_{R,0}(t)$ and $\delta_T(t) = \beta_{0,T}(t)$.

Multiplexors are multi-input-single-output (MISO) service elements. A multiplexer has a total service curve $\beta(t)$ which can be offered to the aggregate of the passing flows. A service curve offered to a particular flow is a portion of the total service $\beta(t)$.

Arbitrary multiplexer, also referred to as first-come-first-serve (FCFS) in [14] or FIFO multiplexer, does not favour any flow over another. Therefore, in worst case, it must be considered that the service curve β^{F_1} is the leftover after serving the arbitrating flow F_2 . The following theorem is a modification of the Corollary 2.4.1. in [10, 106]:

Theorem 2.1 (FCFS Multiplexer Service Curve) *Let us have a FCFS multiplexer with a total service curve $\beta_{R,T}(t)$ passed through by the flows F_1 and F_2 . The service curve offered to the flow F_1 is $\beta^{F_1}(t) = [\beta(t) - \alpha_2(t)]^+$. Provided that the flow F_2 has γ_{r_2, b_2} as an arrival curve, the service curve offered to the flow F_1 is*

$$\beta^{F_1}(t) = (R - r_2) \left[t - \frac{RT + b_2}{R - r_2} \right]^+ = \beta_{R-r_2, \frac{RT+b_2}{R-r_2}}(t)$$

Strict priority (SP) multiplexer favours one flow absolutely over the other. However, there is an additional latency caused by the fact that a packet of the low-priority flow is being processed at the moment of arrival of a packet of the high priority flow. Therefore, the packet must wait until the low-priority packet is served. The service curve was extended to multiple inputs in [20].

Theorem 2.2 (SP Multiplexer Service Curve) *Let us have a SP multiplexer with a total service curve $\beta_{R,T}(t)$ passed through by a flows $F_i(t), i = 1..n$ bounded by arrival curve $\alpha_i(t)$, where the priority decreases with increasing i . The service curve offered to the flow $F_i(t)$ is*

$$\beta^{F_i} = \left[\beta(t) - l_{max}^L - \sum_{0 < j < i} \alpha_j(t) \right]^+, \quad i = 1, \dots, n-1, \quad \beta^{F_n} = \left[\beta(t) - \sum_{0 < j < n-1} \alpha_j(t) \right]^+$$

where l_{max}^L is the maximum packet length among all flows. Provided that the flows F_i are constrained by $\alpha_i(t) = \gamma_{r_i, b_i}(t)$ and the service curve offered to the aggregate of the flows is $\beta(t) = \beta_{R,T}(t)$, service curve offered to the flow F_i is

$$\begin{aligned} \beta^{F_i} &= \left(R - \sum_{0 < j < i} r_j \right) \left[t - \frac{RT + \sum_{0 < j < i} b_j}{R - \sum_{0 < j < i} r_j} \right]^+, \quad i = 1, \dots, n-1, \\ \beta^{F_n} &= \left(R - \sum_{0 < j < n-1} r_j \right) \left[t - \frac{RT + \sum_{0 < j < n-1} b_j}{R - \sum_{0 < j < n-1} r_j} \right]^+ . \end{aligned}$$

Le Boudec introduces in [10, 28] the following three theorems representing the basic performance bounds: backlog bound, delay bound and the output bound. Backlog bound retrieves the size of the buffer needed in the service node to be able to process the incoming flow. Delay bound represents the maximum delay of a bit passing through a service element given the arrival curve. The output bound represents the arrival curve which constrains the outgoing traffic from the service element.

Theorem 2.3 (Backlog Bound) *Assume a flow, constrained by arrival curve α , traverses a system that offers a service curve β . The backlog $R(t) - R^*(t)$ for all t satisfies $R(t) - R^*(t) \leq \sup_{s \geq 0} \{\alpha(s) - \beta(s)\}$.*

Theorem 2.4 (Delay Bound) *Assume a flow, constrained by arrival curve α traverses a system that offers a service curve β . The maximum delay $\hat{d}(t)$ for all t satisfies: $\hat{d}(t) \leq h(\alpha, \beta)$, where*

$$h(\alpha, \beta) = \sup_{s \geq 0} \{ \inf_{\tau \geq 0} \{ \alpha(s) \leq \beta(s + \tau) \} \}.$$

Theorem 2.5 (Output Bound) *Assume a flow, constrained by arrival curve α , traverses a system that offers a service curve of β . The output flow is constrained by the arrival curve $\alpha^* = \alpha \odot \beta$.*

The operations \otimes and \oslash represent the min-max convolution and deconvolution respectively. Details and proofs can be found in [10].

After describing the basic service elements a method must be chosen to concatenate these elements. There are several approaches. Schmitt et al provided a classification of the available approaches in [32]. In Total Flow Analysis (TFA), the end-to-end delay is then equal to the sum of the elements' delays along the flow's path. Disadvantageous of this method is that only end-to-end delay bound can be obtained.

Separate Flow analysis SFA is based on the assumption that the service offered to the observed flow is given by concatenation of the partial service curves offered to the flow at each service element. The following theorem is based on Theorem 1.4.6 in [10].

Theorem 2.6 (Service Curve Concatenation) *Let us have two systems S_1 and S_2 with service curves β_1 and β_2 , respectively. The system S formed by concatenation of the systems S_1 and S_2 offers the flow a service curve $\beta(t) = \beta_1(t) \otimes \beta_2(t)$.*

Furthermore, it can be shown that for rate-latency service curves the following holds:

Theorem 2.7 (Rate-Latency Service Curve Concatenation) *Let us have two systems S_1 and S_2 with service curves $\beta_1(t) = \beta_{R_1, T_1}(t)$ and $\beta_2(t) = \beta_{R_2, T_2}(t)$, respectively. The system S formed by concatenation of the systems S_1 and S_2 offers the flow a service curve $\beta(t) = \beta_{\min\{R_1, R_2\}, T_1 + T_2}(t)$.*

Pay Multiplexing Only Once (PMOO-SFA) is an extension of the SFA. The core idea is that burst caused by the cascaded FCFS aggregation can be paid only once. Hence, in an example of two concatenated FCFS nodes given by β_1 and β_2 into which two flows are multiplexed (α_1 and α_2), while we observe α_1 and consider α_2 as a competing flow. The PMOO approach considers the following statement valid

$$[(\beta_1 \otimes \beta_2) - \alpha_2]^+ \approx [\beta_1 - \alpha_2]^+ \otimes [\beta_2 - (\alpha_2 \oslash \beta_1)]^+.$$

Details on PMOO are introduced in [32] and [33].

Finally, the main idea of Extended PBOO (EPBOO) is that if all flows are $\gamma_{r,b}$ -constrained and all service elements of the path are of the $\beta_{R,T}$ -type then the end-to-end service curve is also the $\beta_{R,T}$ -type such that

$$\beta_i(t) = \min_{j \in \mathbb{J}_i} \left[R^j - \sum_{k \in \mathbb{K}^j} r_k \right] \cdot \left[t - \sum_{j \in \mathbb{J}_i} T^j - \sum_{k \in \mathbb{K}_i} \frac{b_k^{j, \min}}{\min_{j \in \mathbb{J}_{i,k}} [R^j]} \right]^+, \quad (1)$$

where i is the flow index, j is the index of the service element $\beta_{R,T}^j$ along the path and k is the index of flows interfering the flow i at service element j . \mathbb{J}_i is the set of service elements of flow i , \mathbb{K}^j is the set of flows interfering with the flow i at service element j , $\mathbb{K}_i = \bigcup_{j \in \mathbb{J}_i} \mathbb{K}^j$ is the set of all interfering flows along the path of flow i , and $\mathbb{J}_{i,k} = \mathbb{J}_i \cap \mathbb{J}_k$ is the set of service elements used both by flow i and flow j .

There are several more issues complementary to modelling of real-world systems which are introduced in the dissertation.

3 EMPIRICAL ANALYSIS OF PERFORMANCE BOUNDS

Empirical analysis of performance bounds is important for finding QoS parameters of interest and for retrieval of dependencies of QoS parameters on different combination of flows, networking device parametrisation, and networking devices themselves.

3.1 TestQoS: Quality of Service Test Bed

The QoS parameters of interest are given by QoS metrics introduced in Section 2.4. The most important metrics are *packet latency* which can be measured directly, and *latency jitter*, which can be inferred from repetitive measurements. From Section 2.2 one can understand that sub-millisecond resolution of packet latency is required in order to obtain sensible measurement. Such a temporal resolution can be obtained either by specialised hardware test beds, or by real-time extension of an operating system providing precise clock signals. Generic operating systems with Ethernet NICs are usually not capable of delivering acceptable precision.

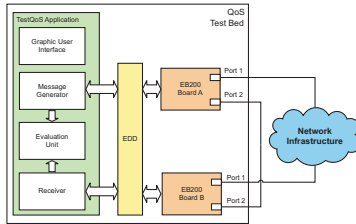


Figure 4: TestQoS architecture

Figure 4 shows the TestQoS architecture [6]. It is based on two Siemens EB200 boards which is suitable for further extensions providing packet latency measurements over larger distances. *Graphical User Interface* is used for test flow parametrisation and test progress control. *Message Generator* generates the test traffic which is transmitted via *Port 1* and *Port 2* of *Board A*. The traffic is received by *Receiver*. Latency for every packet pair is calculated in *Evaluation Unit*. The raw data are then processed and visualised.

Packet latency measurement employs the following approach. Ports 1 are connected over the measured network infrastructure and Ports 2 are connected by a patch cable forming a shortcut. Capturing counter values proceeds in the following manner: i -th packet is transmitted via *Port 1* of *Board A*, timestamp t_i^{Tx1} is captured. Duplicated i -th packet is transmitted via *Port 2* of *Board A*, timestamp t_i^{Tx2} is captured. i -th packet is received by *Port 1* of *Board B*, timestamp t_i^{Rx1} is captured. Duplicated i -th packet is received by *Port 2* of *Board B*, timestamp t_i^{Rx2} is captured. The latency of the i -th packet is

$$d_i = t_i^{Rx1} - t_i^{Tx1} + t^{off}(t_i^{Tx1}), \quad (2)$$

where $t^{off}(t_i^{Tx1})$ is the offset between the values of the free-running counters at the moment of packet transmission. The offset is not guaranteed to be constant. Therefore, it has to be considered as a variable. As the packet latency between Ports 2 is negligible (for 0.5 m long cable, the latency is 1 ns), it holds that

$$t_i^{Rx2} - t_i^{Tx2} + t^{off}(t_i^{Tx2}) = 0. \quad (3)$$

Installing (3) into (2) results in

$$d_i = \lim_{t^{off}(t_i^{Tx2}) \rightarrow t^{off}(t_i^{Tx1})} \{t_i^{Rx1} - t_i^{Tx1} - t_i^{Rx2} + t_i^{Tx2}\}. \quad (4)$$

3.2 Measurement Results

This chapter introduces a set of test cases with the HP ProCurve 1800-8G switch and the Cisco 2811 router used further for modelling. These four main test cases provide evidence of SF blocking or non-blocking nature, and the behaviour with the congested OI under different scheduling mechanism. Each test case is represented by a test topology and a figure with the resulting dependency. Details of the measurements are documented in the dissertation.

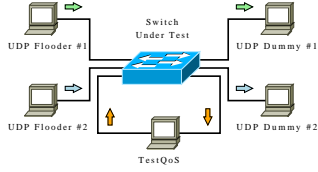


Figure 5: Test case: switch-fabric loading at switch (SW.FL) - topology and maximum latency. Additional traffic has no influence, max. lat. is constantly 90.5 μ s

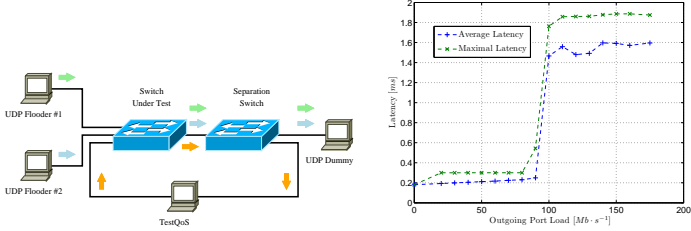


Figure 6: Test case: outgoing port congestion at switch (SW.OPC) - topology and maximum latency

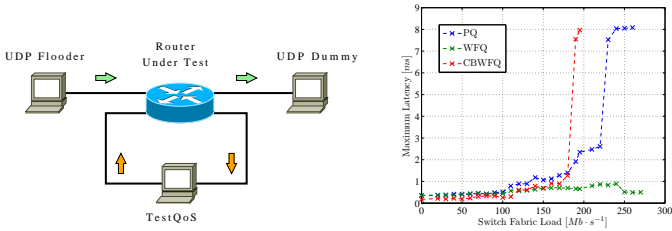


Figure 7: Test case: switch-fabric loading at router (RTR.FL) - topology and maximum latency

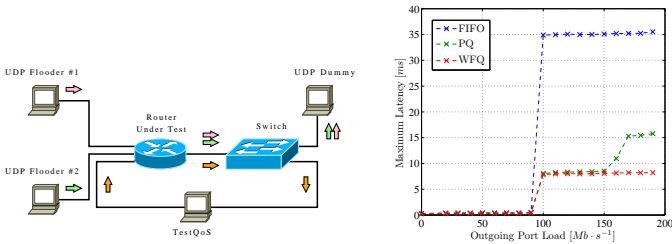


Figure 8: Test case: outgoing port congestion at router (RTR.OPC) - topology and maximum latency

4 NETWORKING DEVICE MODELLING

4.1 Rate-Variable-Latency Service Curve

In cases shown in the *SW.OPC* and *RTR.OPC*, it is difficult to model the observed latency dependency with the means available in the network calculus framework; test cases show behaviour in which the packet latency increases step-wise despite the highest priority. Such a behaviour corresponds to a service curve with bimodal latency parameter shown in Figure 9. In such a case the rate-latency service curve becomes rate-variable-latency (RVL) service curve and has a form $\beta_{R,T_1,T_2,r_T}(\alpha(t), t)$. RVL is dependent on the parameters of the incoming flow $F(t)$, similarly to multiplexors.

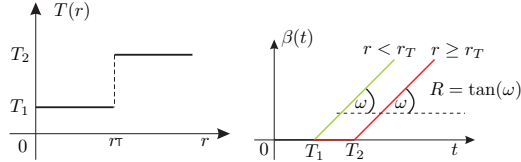


Figure 9: Service curve parameter T vs. arrival curve parameter r (left), and Rate-variable-latency service curve (right)

Definition 4.1 (RVL Service Curve Function) Let us have a rate-latency service curve $\beta_{R,T}(t) = R[t - T]^+$ with constant parameters of rate R and latency T , and an arrival curve $\alpha(t) = \gamma_{b,r}(t) = rt + b$. If the latency T of the service curve is dependent on the parameter r of the arrival curve $\alpha(t)$ such that $T = \{T_1, r < r_T; T_2, r \geq r_T\}$, where the parameter r_T is a trigger rate, then the service becomes a Rate-Variable-Latency (RVL) service curve and is defined as

$$\beta_{R,T_1,T_2,r_T}(\alpha(t), t) = R[t - T_1 - (T_2 - T_1)\mathbf{1}_{\{\alpha(1) - \alpha(0) \geq r_T\}}]^+. \quad (5)$$

The step function is used $v_T(t) = \mathbf{1}_{\{t \geq T\}}$ defined in [10, 129]. Furthermore, if the arrival curve is affine curve, it holds that $r = \alpha(1) - \alpha(0)$. In some cases the following form of the RVL function is more suitable

$$\beta_{R,T,T_{con},r_T}(\alpha(t), t) = R[t - T_1 - T_{con}\mathbf{1}_{\{\alpha(1) - \alpha(0) \geq r_T\}}]^+, \quad (6)$$

4.2 Definition of Model Structure

Figure 2 shows the general hardware architecture of a networking device. In spite of the fact that the hardware architecture is composed of three stages, the networking device model was decided to be two-stage. The former service curve represents incoming interfaces together with the SF, and the latter service curve represents the OI as shown in Figure 10.

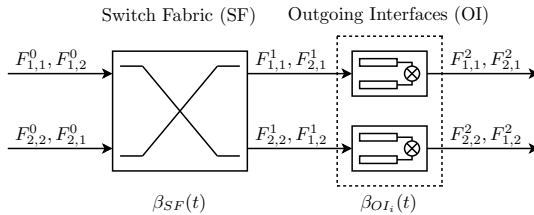


Figure 10: Port-to-port service offered to flows

$F_{i,j,k}^l$ denotes a flow entering the device via interface i to be forwarded via interface j and belonging to the priority class k , if applicable. The upper index l denotes the internal hop. Entering flow is upper-indexed as 0. $\beta_{se}^{F_{i,j,k}}$ denotes a service curve of the service element se offered to the flow $F_{i,j,k}$. The same indexing applies also to the service curve parameters. So, if $\beta_{se}^{F_{i,j,k}} = \beta_{R,T}$, then $R = R_{se}^{F_{i,j,k}}$, and $T = T_{se}^{F_{i,j,k}}$.

4.2.1 Switch Fabric Model Structure

There are two aspects important for inferring a representative SF model: *blocking property* and *scheduling algorithm*.

Blocking has a close connection to the SF type and buffer location. Non-blocking means that any input/output pair of ports of the switch pairs can be connected if neither of the ports is occupied [11, 179]. For the analysis, this means that latency remains unchanged under load. Hence, let us establish a coarse classification of SF models based on this feature. If the SF is blocking the SF-B constellation in Figure 11 is recommended to be used. In case of a non-blocking SF, constellation SF-N in Figure 12 can be employed.

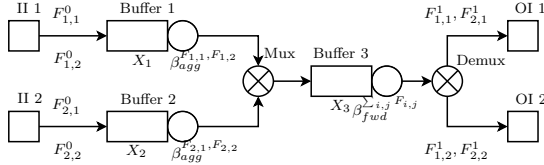


Figure 11: Blocking switch fabric (SF-B)

Figure 11 shows a general SF-B architecture counting all possible blocks. Buffer 1 and 2 represent optional input buffer. Mux represents the scheduler polling the packets and passing them to SF. The rate of the SF is given by the shared-medium capacity. Buffer 3 represents the optional buffer. The stored packets are served in FCFS order and forwarded to the OI. The rate of the forwarding unit is given by its forwarding capacity and the latency is the implicit forwarding overhead.

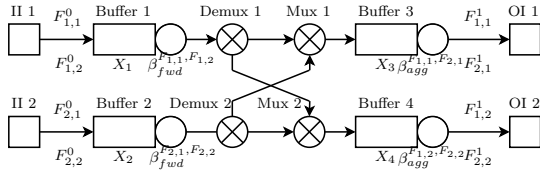


Figure 12: Non-blocking switch fabric (SF-N)

Figure 12 shows a general SF-N architecture counting all possible blocks. Buffer 1 and 2 represent input buffers and the rate-latency nature of the output is given by the forwarding unit as in the SF-B case. Buffer 3 and 4 represent the output buffers needed in case of congestion of the OI. The rate of the buffer output is given by the bus rate connected to the OI of the networking device.

If the SF architecture is not known, discovering the SF architecture at this level of granularity is straightforward. Using the test cases SW.FL or RTR.FL, it can be decided if the implementation is SF-based with multi-path architecture, or over-provisioned bus-based architecture on one hand, or bus-based imposing forwarding limitations. This coarse classification is sufficient.

However, more heuristics is necessary to retrieve the scheduling mechanism of the SF. The discovery is based on the test scenario introduced in the SW.FL and RTR.FL. The principle is based on different loading behaviour under different combinations of the rates and priorities of the flows used for the testing. Details are introduced in the dissertation. Finally the service curve offered to the flow $F_{i,j,k}^l(t)$ by the SF under the PBOO assumption can be expressed as

$$\beta_{SF}^{F_{i,j,k}^l}(t) = (\beta_{agg}^{F_{i,j,k}} \otimes \beta_{fwd}^{F_{i,j,k}})(t), \quad (7)$$

where $\beta_{agg}^{F_{i,j,k}}$ represents the aggregation of flows being forwarded to the common OI and the $\beta_{fwd}^{F_{i,j,k}}$ represents the forwarding resolution for the given flows. Table 1 shows the forms of the service curves and the flows which are aggregated in the given service elements. One can observe that SF-B constellation is subject to aggregation of more flows. The table introduces only the types of the curves. Specific parameters are based on the scheduling mechanisms employed. The most common scheduling mechanisms are FCFS, and RR.

SF Type	$\beta_{agg}^{F_{i,j}}$	$\beta_{fwd}^{F_{i,j}}$
SF-B	$\lambda_R, R = R_{agg}^{\sum_j F_{i,j}}$	$\beta_{R,T}, R = R_{fwd}^{\sum_j F_{i,j}}, T = T_{fwd}^{\sum_j F_{i,j}}$
SF-N	$\lambda_R, R = R_{agg}^{\sum_j F_{i,j}}$	$\beta_{R,T}, R = R_{fwd}^{\sum_j F_{i,j}}, T = T_{fwd}^{\sum_j F_{i,j}}$

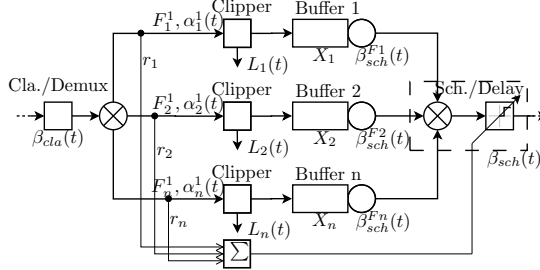


Figure 13: Model of an OI

4.2.2 Outgoing Interface Model Structure

OI can be modelled with a rate-latency service curve $\beta_{OI} = \beta_{R,T}(t)$. Its actual form depends on the scheduler type, number of queues, bandwidth allocations (if applicable), and queue lengths.

The results of the test cases SF.OPC and RTR.OPC show that additional latency occurs when the OI becomes congested. The state switching is obviously controlled by the total data rate and the trigger is equal to the link capacity. The theoretical background of this behaviour is given in [7].

While SF can be prevented from data loss by over-provisioning, there is no such possible treatment at OIs where service rate is limited by the interface capacity. Consequently, data loss is an intrinsic part of the OI. [10, 251] gives a basic theorem evaluating the data loss. However, within this work, use of the closed-form results inferred for this purpose within the dissertation is made.

The block diagram of an exemplary OI is shown in Figure 13. The interface has three queues representing either flows or classes of flows depending on the chosen congestion management mechanism. The packets arriving from the SF are demultiplexed if congestion management is employed. Demultiplexing is based on a decision of a classifier which analyses the packet content and forwards the packet to the respective queue. The processing capacity of the classifier is given by $\beta_{cla}(t)$. Clippers represent data loss. The flow $L_i(t)$ is a part of the flow $F_i(t)$ which is discarded in order that the residual flow can traverse the rest of the system with finite buffer of X_i in a lossless manner. The packets advance to queues and are served by the scheduler based on the chosen mechanism. Hence, the total service capacity of $\lambda_R(t)$ is split to $\lambda_{RF_i}(t)$ accordingly. Finally, the forwarding service element represents the overhead of the OI caused by the processing delays. The latency depends on the OI congestion condition given by the trigger r_T . T_1 represents the contention-free latency and T_2 represents the congested latency. As a result a service offered to a flow $F_i(t)$ by the OI is

$$\beta_{OI}^{F_i} \left(t, \sum r_i \right) = \left(\beta_{cla}^{F_i} \otimes \beta_{sch}^{F_i} \right) \left(t, \sum r_i \right), \quad (8)$$

where the service curves have parameters expressed by Table 2.

β_{cla}	$\lambda_R, R = R_{cla}$
β_{sch}	$\beta_{R,T}, R = R_{agg}^{F_i}, T = T_{sch} + T_{sch,con} \cdot 1 \left\{ \sum_{i=1}^n r_i \geq r_T \right\}$

The loss rate bound of the flow $F_i(t)$ bounded by $\alpha_i = \gamma_{r_i, b_i}$ experienced at the OIs is bounded by

$$\hat{t}^{F_i}(t) = \left[1 - \inf_{0 < s \leq t} \frac{\left(\beta_{cla}^{F_i} \otimes \beta_{sch}^{F_i} \right) (s, \sum r_i) + X_i}{\alpha_i(s)} \right]^+. \quad (9)$$

4.3 Port-to-Port Service Curve of a Networking Device

So far, modelling of SF and OI has been accomplished in Section 4.2. Due to traffic aggregation and deaggregation, it is worth considering which approach to take to obtain the port-to-port service curve of a modelled device. The objective is to find the most suitable method to obtain port-to-port service curve. Available approaches are introduced in Section 2.5. Two approaches will be applied: (i) PBOO with ad hoc application of PMOO where possible, and (ii) Extended PBOO. Initial experiment is introduced in [8]. It is supposed that there is no loss at the SF. The only data loss is allowed at the OIs. The scheduling algorithm at the SF is arbitrary (FCFS).

4.3.1 PBOO/PMOO Approach

The PBOO mechanism is based on the assumption that the port-to-port service curve offered to a flow is a convolution of the service curves offered to the traversing flow. The service curves withal represent the service elements which the flow traverses. This principle is introduced in Section 2.5 and is given by (10).

$$\beta_{p2p}^{F_{I,J,K}} = \bigotimes_{h \in \mathcal{H}} \beta_h^{F_{I,J,K}}, \quad (10)$$

where $F_{I,J,K}$ is the observed flow, h represents the order of the service element (hop) traversed by the flow, and \mathcal{H} represents the set of hops on the flow's path. If applicable, the principle can be extended by the PMOO principle also introduced in Section 2.5.

$$\beta_{h,h+1}^{F_{I,J,K}} = [(\beta_h \otimes \beta_{h+1}) - \alpha_{i,j,k}]^+ \approx [\beta_h - \alpha_{h+1}]^+ \otimes [\beta_{h+1} - (\alpha_{i,j,k} \circ \beta_{i,j,k})]^+. \quad (11)$$

In the following, port-to-port service curves offered to the observed flow will be inferred for the two finally validated device architectures, i.e., SF-B/OI-PQ and SF-N/OI-PQ.

With the SF-B/OI-PQ, one can use the PMOO with the SF-B architecture as all flows pass the same part of the subpath. However, it no longer applies for the OI as the flows forwarded to different OI deaggregate. From Theorem 2.2, it holds that the observed flow competes only with flows of higher priority, and potentially has to wait for completion of transmission of any lower-priority flow. These facts are regarded in (12).

$$\begin{aligned} \beta_{p2p}^{F_{I,J,K}} &= \beta_{SF}^{F_{I,J,K}} \otimes \beta_{OI}^{F_{I,J,K}}, \\ \beta_{SF}^{F_{I,J,K}} &= \left[(\beta_{agg} \otimes \beta_{fwd}) - \sum_{\{i,j,k\} \in \mathbb{S}_1} \alpha_{i,j,k}^0 \right]^+, \\ \beta_{OI}^{F_{I,J,K}} &= \left[\beta_{cla} - \sum_{\{i,j,k\} \in \mathbb{S}_2} \alpha_{i,j,k}^1 \right]^+ \otimes \left[\beta_{sch} - \sum_{\{i,j,k\} \in \mathbb{S}_3} \alpha_{i,j,k}^2 - \{L_{max}^I\}_{K < K_{max}} \right]^+. \end{aligned} \quad (12)$$

The service curve is based on FCFS and PQ multiplexors with service curves in Theorem 2.1 and Theorem 2.2. The sets are the following:

$$\mathbb{S}_1 = \{[i, j, k]; \cup - [I, J, K]\}, \mathbb{S}_2 = \{[i, j, k]; j = J\} - [I, J, K], \mathbb{S}_3 = \{[i, j, k]; j = J, k \leq K\} - [I, J, K].$$

Finally, additional flows' bounds $\alpha_{i,j,k}^1 \in \mathbb{S}_2$ and $\alpha_{i,j,k}^2 \in \mathbb{S}_3$ have to be inferred according to the rule $\alpha_{i,j,k}^{h+1} = \alpha_{i,j,k}^h \circ \beta_h^{i,j,k}$.

In case of the SF-N/OI-PQ, PMOO cannot be used at all due to the aggregation scheme. Hence, $\beta_{p2p}^{F_{I,J,K}}$ must be expressed as a concatenation of four service curves according to (13).

$$\begin{aligned} \beta_{p2p}^{F_{I,J,K}} &= \beta_{SF}^{F_{I,J,K}} \otimes \beta_{OI}^{F_{I,J,K}}, \\ \beta_{SF}^{F_{I,J,K}} &= \left[\beta_{fwd} - \sum_{\{i,j,k\} \in \mathbb{S}_1} \alpha_{i,j,k}^0 \right]^+ \otimes \left[\beta_{agg} - \sum_{\{i,j,k\} \in \mathbb{S}_2} \alpha_{i,j,k}^1 \right]^+, \\ \beta_{OI}^{F_{I,J,K}} &= \left[\beta_{cla} - \sum_{\{i,j,k\} \in \mathbb{S}_3} \alpha_{i,j,k}^2 \right]^+ \otimes \left[\beta_{sch} - \sum_{\{i,j,k\} \in \mathbb{S}_4} \alpha_{i,j,k}^3 - \{L_{max}^I\}_{K < K_{max}} \right]^+. \end{aligned} \quad (13)$$

The sets of flow aggregated at the SF and OI are given by the following conditions.

$$\mathbb{S}_1 = \{[i, j, k]; i = I\} - [I, J, K], \quad \mathbb{S}_2 = \mathbb{S}_3 = \{[i, j, k]; j = J\} - [I, J, K], \quad \mathbb{S}_4 = \{[i, j, k]; j = J, k \leq K\} - [I, J, K].$$

Finally, it is necessary to infer the bounds of the aggregated flow passing through the observed OI to complete the knowledge of all elements in (13). $\alpha_{i,j,k}^h$, $h = 1, 2, 3$ can be obtained according the same approach as above.

4.3.2 Extended PBOO Approach

Extended PBOO (EPBOO) is based on stronger assumptions delivering tighter bounds, e.g., multiplexers are based on arbitrary scheduling. As the general assumption of this work is that the SF apply arbitrary scheduling, EPBOO can be not only applied with OIs with FCFS scheduler, yet for OIs with PQ after a small consideration.

$$R_{p2p}^{F_{I,J,K}} = \min_{h \in \mathbb{J}(I,J,K)} \left[R_h - \sum_{\{i,j,k\} \in \mathbb{K}^h} r_{i,j,k} \right], \quad (14)$$

$$T_{p2p}^{F_{I,J,K}} = \sum_{h \in \mathbb{J}(I,J,K)} T_h + \sum_{\{i,j,k\} \in \mathbb{K}(I,J,K)} \frac{b_{i,j,k}^{h_{min}}}{\min_{h \in \mathbb{J}(I,J,K), \{i,j,k\}} [R_h]}. \quad (15)$$

The following consideration has to be made in order that EPBOO is applicable to the SF-B/OI-PQ device architecture. The proposition is based on Theorem 2.1 and Theorem 2.2. The idea is that the observed flow competes with all flows with $k \leq K$ in an arbitrary manner. Consequently, all flows with $k > K$ can be neglected. Furthermore, the observed flow has to wait for finalisation of transmission of the current packet. Hence, the observed flow has to wait for $\frac{L_{max}}{R_{OI}}$ if $K < K_{max}$. The resulting parameters of $\beta_{p2p}^{F_{I,J,K}}$ are

$$R_{p2p}^{F_{I,J,K}} = \min \left[\min[R_{agg}, R_{fwd}] - \sum_{\mathbb{S}_{R_1}} r_{i,j,k}, R_{cla} - \sum_{\mathbb{S}_{R_2}} r_{i,j,k}, R_{sch} - \sum_{\mathbb{S}_{R_3}} r_{i,j,k} - \sum_{\mathbb{S}_{R'_5}} r'_{i,j,k} \right], \quad (16)$$

$$\begin{aligned} T_{p2p}^{F_{I,J,K}} &= T_{fwd} + T_{sch} + \left\{ \frac{L_{max}}{R_{OI}} \right\}_{K < K_{max}} + \frac{\sum_{\mathbb{S}_{B_1}} b_{i,j,k}}{\min[R_{agg}, R_{fwd}]} + \frac{\sum_{\mathbb{S}_{B_2}} b_{i,j,k}}{\min[R_{agg}, R_{fwd}, R_{cla}, R_{sch}]} + \\ &+ \frac{\sum_{\mathbb{S}_{B_3}} b_{i,j,k}}{\min[R_{agg}, R_{fwd}, R_{cla}]} + \frac{\sum_{\mathbb{S}_{B_3}'} b'_{i,j,k}}{R_{sch}}. \end{aligned} \quad (17)$$

The conditions which hold for different sets introduced in (16) and (17) are evident from Figure 11. Thus,

$$\begin{aligned} \mathbb{S}_{R_1} &= \{[i, j, k]; \cup - [I, J, K]\}, \quad \mathbb{S}_{R_2} = \{[i, j, k]; j = J\} - [I, J, K], \\ \mathbb{S}_{R_3} &= \{[i, j, k]; j = J, k = K\} - [I, J, K], \quad \mathbb{S}_{R'_5} = \{[i, j, k]; j = J, k < K\}, \\ \mathbb{S}_{B_1} &= \{[i, j, k]; j \neq J\} \cup \{[i, j, k]; j = J, k \neq K\}, \\ \mathbb{S}_{B_2} &= \{[i, j, k]; j = J, k = K\} - [I, J, K], \quad \mathbb{S}_{B_3} = \{[i, j, k]; j = J, k < K\}. \end{aligned}$$

By applying the same consideration as with the SF-B/OI-PQ architecture, the resulting parameters of the service curve $\beta_{p2p}^{F_{I,J,K}}$ for the SF-N/OI-PQ architecture are

$$R_{p2p}^{F_{I,J,K}} = \min \left[R_{fwd} - \sum_{\mathbb{S}_{R_1}} r_{i,j,k}, R_{agg} - \sum_{\mathbb{S}_{R_2}} r_{i,j,k}, R_{cla} - \sum_{\mathbb{S}_{R_3}} r_{i,j,k}, R_{sch} - \sum_{\mathbb{S}_{R_4}} r_{i,j,k} - \sum_{\mathbb{S}_{R'_4}} r'_{i,j,k} \right], \quad (18)$$

$$\begin{aligned} T_{p2p}^{F_{I,J,K}} &= T_{SF} + T_{OI} + \left\{ \frac{L_{max}}{R_{OI}} \right\}_{K < K_{max}} + \frac{\sum_{\mathbb{S}_{B_1}} b_{i,j,k}}{R_{fwd}} + \frac{\sum_{\mathbb{S}_{B_2}} b_{i,j,k}}{\min[R_{fwd}, R_{agg}, R_{cla}]} + \frac{\sum_{\mathbb{S}_{B_3}} b_{i,j,k}}{\min[R_{agg}, R_{cla}]} + \\ &+ \frac{\sum_{\mathbb{S}_{B_4}} b_{i,j,k}}{\min[R_{agg}, R_{cla}, R_{sch}]} + \frac{\sum_{\mathbb{S}_{B_5}} b_{i,j,k}}{\min[R_{agg}, R_{cla}]} + \frac{\sum_{\mathbb{S}_{B_5}'} b'_{i,j,k}}{R_{sch}}. \end{aligned} \quad (19)$$

Finally, the sets respect the arbitration rules of the PQ at the OI:

$$\begin{aligned}
\mathbb{S}_{R_1} &= \{[i, j, k]; i = I\} - [I, J, K], & \mathbb{S}_{R_2} &= \{[i, j, k]; j = J\} - [I, J, K], \\
\mathbb{S}_{R_3} &= \{[i, j, k]; j = J\} - [I, J, K], & \mathbb{S}_{R_4} &= \{[i, j, k]; j = J, k = K\} - [I, J, K], \\
\mathbb{S}'_{R'_4} &= \{[i, j, k]; j = J, k < K\}, & \mathbb{S}_{B_1} &= \{[i, j, k]; i = I, j \neq J\}, \\
\mathbb{S}_{B_2} &= \{[i, j, k]; i = I, j = J, k \neq K\}, & \mathbb{S}_{B_3} &= \{[i, j, k]; i \neq I, j = J, k \neq K\}, \\
\mathbb{S}_{B_4} &= \{[i, j, k]; i \neq I, j = J, k = K\}, & \mathbb{S}_{B_5} &= \{[i, j, k]; j = J, k < K\}.
\end{aligned}$$

4.4 Identification of the Model Parameters

In case of lack of device information, it is necessary to perform parameter identification in order to have a valid model of the networking device. The principal problem is that any model based on network calculus operates with bounds, i.e., the worst case situations while empirical analysis is subject to randomness. Hence, special attention has to be paid to measurement stability and confidentiality. Moreover, not all parameters are measurable and thus some parameters have to be merged even at cost of deteriorated precision; for instance T_{SF} and T_{OI} appear in a sum of $T_{SF} + T_{OI}$.

4.4.1 Switch-Fabric Parameters

SF-B is considered in this section. Figure 14 shows two flows used for the identification. F_T is the flow generated and captured in the TestQoS application. F_L is additional loading flow generated by the UDPFlooder or any other application with controllable traffic shape.

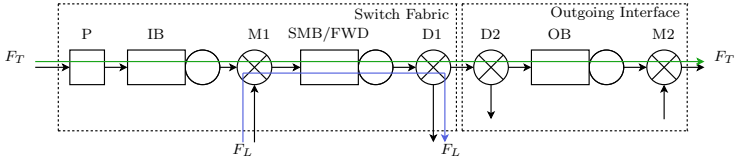


Figure 14: Flows during SF-B identification

The only common part of the path is the SF. The end-to-end service curve offered to the flow F_T under the PBOO assumption is

$$\beta_{p2p}^{F_T} = \beta_{SF}^{F_T} \otimes \beta_{OI}^{F_T}. \quad (20)$$

It holds from (1) that $R_{SF} = \min[R_{agg}, R_{fwd}]$. However, these parameters cannot be recognised by identification. Hence, R_{SF} is used. As the SF is shared by both flows, the service curve offered to F_T is equal to

$$\beta_{SF}^{F_T} = [\beta_{SF} - \alpha_L]^+ = (R_{SF} - r_L) \left[t - \frac{R_{SF}T_{SF} + b_L}{R_{SF} - r_L} \right]^+. \quad (21)$$

Similarly, OI is used only by F_T . Hence,

$$\beta_{OI}^{F_T} = \beta_{OI} = R_{OI} [t - T_{OI}]^+. \quad (22)$$

Inserting (21) and (22) into (20) yields in

$$\beta_{p2p}^{F_T} = \min[(R_{SF} - r_L), R_{OI}] \cdot \left[t - \frac{R_{SF}T_{SF} + b_L}{R_{SF} - r_L} - T_{OI} \right]^+. \quad (23)$$

The bound of virtual latency \hat{d}_{F_T} from Theorem 2.4 is

$$\hat{d}_{F_T} = T_{p2p}^{F_T} + \frac{b_T}{R_{p2p}^{F_T}}. \quad (24)$$

Finally, by inserting (23) into (24) yields

$$\hat{d}_{F_T} = \frac{R_{SF}T_{SF} + b_L}{R_{SF} - r_L} + T_{OI} + \frac{b_T}{\min[(R_{SF} - r_L), R_{OI}]}. \quad (25)$$

(24) is of importance as the port-to-port latency is the only measurable latency. As the most important question is what the influence of flow aggregation is, the dependence $\hat{d}_{F_T} = f(F_L)$ is observed and used for identification.

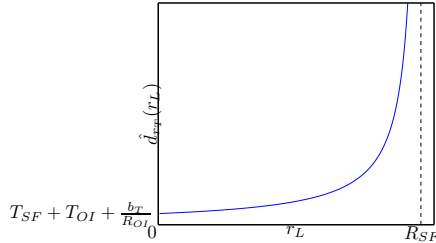


Figure 15: SF-B parameter identification

Figure 15 shows the dependence $\hat{d}_{F_T} = f(r_L)$. It implies from (24) that setting r_L to zero results in

$$\hat{d}_{F_T}(r_L = 0) = T_{SF} + T_{OI} + \frac{b_T}{R_{OI}}.$$

The b_T parameter is known for being generated in TestQoS. $\hat{d}_{F_T}(r_L = 0)$ is the measured delay on an offloaded device. Provided that R_{OI} is known the $T_{SF} + T_{OI}$ can be inferred directly. Furthermore, from the shape of the dependence $\hat{d}_{F_T} = f(r_L)$, the R_{SF} is the asymptote of the dependence. In a SF with an infinite buffer, $\lim_{r_L \rightarrow R_{SF}} \hat{d}_{F_T} = \infty$. The following procedure is proposed to find R_{SF} .

1. In a topology with a flow setup introduced in Figure 15, $\hat{d}_{F_T}(r_L)$ must be measured for a number of values of r_L . The more measurements the finer will be the extrapolation.
2. Provided that $T_{OI} \ll T_{SF}$ and b_T is sufficiently small, (25) can be expressed in the following simpler form

$$\hat{d}_{F_T} \approx \frac{R_{SF}(T_{SF} + T_{OI}) + b_L + b_T}{R_{SF} - r_L} = \frac{A}{B + r_L}. \quad (26)$$

The first assumption is reachable and for the given range of investigations, the error can be mitigated. The second assumption can be controlled in such a way the b_T is minimised by introducing short packets.

3. Hyperbolic regression is involved which provides the required parameters A and B in (26). Application of hyperbolic regression is described in [30]. Hence,

$$R_{SF} = -B, \quad T_{SF} + T_{OI} = \frac{A + b_T + b_L}{B}. \quad (27)$$

4.4.2 Outgoing Interface Parameters

Similar topology is used for the OI identification. F_T is the flow generated and captured in the TestQoS application. F_L is additional loading flow generated by the UDPFlooder or any other application with controllable traffic shape. The flow aggregates with F_T in the SF and continues to the same OI.

T_{OI} parameter has been made a part of the common latency parameter within the SF identification and needs no additional treatment. R_{sch} parameter can be identified directly or is known explicitly by the physical capacity of the OI.

According to Section 4.1 where the RVL function was introduced, the scheduler at OI has bimodal latency parameter; a fixed part T_{sch} and the additional part $T_{sch,con}$ caused by congestion. Seeing the fact that the change to different mode is triggered at the OI, it is not possible to incorporate $T_{sch,con}$ into SF. Hence, let T_{sch} be the part of T_{OI} and let the congestion overhead $T_{sch,con}$ be part of the OI. Finally, the task of the OI identification is to find $T_{sch,con}$. Detailed approach is introduced in the dissertation.

5 VALIDATION OF MODELS OF NETWORKING DEVICES

This section provides validation of the derived latency models of the networking devices. In the first validation case, parameterised model of a maximum latency of a switch is inferred and the behaviour is validated on a topology consisting of two switches. The second validation case is dedicated to a router. A parameterised model of maximum latency of the router is inferred and the behaviour is validated on a topology consisting of one switch and one router.

5.1 HP ProCurve 1800-8G Switch

The switch is a configurable Hewlett-Packard 8-port switch with full-duplex interfaces with link capacity of $1 \text{ Gb} \cdot \text{s}^{-1}$. Despite its compact design the switch is extremely time efficient as can be seen in the switch-related measurements in Section 3.2.

Seeing the fact that the capacity of the SF equals to the sum of the throughput of the interfaces, the SF is over-provisioned and thus can be considered as non-blocking (modelled as SF-N).

The I/O blocks accommodate advanced QoS architecture both at input and output which differentiates the QoS architecture from many mid-range routers. Although detailed description of the architecture by means of network calculus would be possible, the trade-off between the precision improvement and the complexity favours simple description. Hence, let us ignore deep architectural details and let us limit our a priori information to that necessary to model the switch using SF-N/OI-PQ model architecture. The PQ assumption with the OI is valid if $r_T \ll r_L$ and $r_T \ll R_{OI}$, which is the condition under which the measurements were made ¹. The reasoning for adopting PQ is given by the fact that there is round-robin mechanism deployed in the switch which automatically favours the low-volume flows.

5.1.1 Switch Parametrisation

Based on the aforementioned characteristics, the switch is a SF-N/OI-PQ device with service curve parameters according to (13). The device throughput is equivalent to the sum of the link capacities of the interfaces. Hence, it is to assume that $R_{SF} = R_{fd} = R_{agg} = 8 \text{ Gb} \cdot \text{s}^{-1}$. The distribution of the fixed latencies given by the T parameters along the device cannot be inferred. Hence, for the matter of simplicity the packet latency of the offloaded switch is assigned to SF, i.e., $T_{SF} = 90.5 \mu\text{s}$, which represents the latency of the offloaded switch.

The contribution of the OI to the total packet latency in case of the congested OI is given by Figure 6, i.e.

$$T_{sch,con} = \hat{d}(r_L = 110 \text{ Mb} \cdot \text{s}^{-1}) - \hat{d}(r_L = 80 \text{ Mb} \cdot \text{s}^{-1}) = 1.858 \text{ ms} - 300 \mu\text{s} = 1.558 \text{ ms}. \quad (28)$$

Due to the fact that the latency in the SW.OPC test case after congestion of the OI does not experience any further saturation in the observed range of loading, it can be assumed that $\beta_{cla} = \infty$. Finally, the OI link rate configured at the switch are $R_{OI} = 100 \text{ Mb} \cdot \text{s}^{-1}$.

5.1.2 Port-To-Port Service Curve

For the model validation, a scenario with the switch with the topology shown in Figure 16 was chosen. The time-critical test flow $F_T(t)$ is disturbed by two additional flows: F_{L_1} used for loading of the OI passes along the same path as $F_T(t)$ to the same OI, and F_{L_2} used for SF loading enters the switch by different II traversing the SF and then continuing to a different OI.

According to the PBOO/PMOO principle, the port-to-port service curve of the switch is (13) and the flows belong to the sets as follows:

$$\mathbb{S}_1 = \emptyset, \quad \mathbb{S}_2 = \{\alpha_{L_1}^1\}, \quad \mathbb{S}_3 = \{\alpha_{L_1}^2\}, \quad \mathbb{S}_4 = \emptyset. \quad (29)$$

Note that the flow F_{L_2} imposes no influence due to the SF-N architecture. As $\beta_{cla} = \infty$, \mathbb{S}_3 is not applicable. \mathbb{S}_4 is considered empty due to the condition $r_T \ll r_{L_1}$ and the round-robin principle guarantees strict priority in this case. Taking this distribution of additional flows into account and using Theorem 2.1, the port-to-port service curve is calculated according to

$$\beta_{p2p}^{F_T} = \beta_{SF}^{F_T} \otimes \beta_{OI}^{F_T} = R_{p2p}^{F_T} [t - T_{p2p}^{F_T}]^+, \quad (30)$$

$$\beta_{SF}^{F_T} = \beta_{fd} \otimes [\beta_{agg} - \alpha_{L_1}^0]^+, \quad \beta_{OI}^{F_T} = [\beta_{cla} - \alpha_{L_1}^1]^+ \otimes [\beta_{sch} - l_{max}]^+.$$

¹This assumption is applicable even in a more general sense in industrial automation where the flows with the highest criticality have low volumes

For the sake of simplicity, β_{fwd} shall be absorbed by β_{agg} as it does not add value, yet cannot be investigated separately. Hence, SF can be inferred directly, subject that $\alpha_{L_1}^0 = r_{L_1}t + b_{L_1}$:

$$\beta_{SF}^{Fr} = (R_{SF} - r_{L_1}) \left[t - \frac{R_{SF}T_{SF} + b_{L_1}}{R_{SF} - r_{L_1}} \right]^+ . \quad (31)$$

As $\beta_{cla} = \infty$, $[\beta_{cla}^{Fr} - \alpha_{L_1}^1]^+ = \infty$ as well. As for the convolution operator \otimes is a neutral element, $\beta_{OI}^{Fr} = [\beta_{sch} - l_{max}]^+$. Subsequently, $\alpha_{L_1}^1$ does not have to be calculated. At last, the service curve of the scheduler must be inferred. The service curve is of the RVL type according to (5). Moreover, the F_T is considered of the highest priority and it does not compete with any other traffic. Thus,

$$\beta_{sch}^{Fr} = R_{sch} \left[t - T_{sch,con} \cdot \mathbf{1}_{\{r_{L_1} + r_T \geq R_{sch}\}} + \frac{l_{max}}{R_{sch}} \right]^+ . \quad (32)$$

Finally, by inserting (31) and (32) into (30), the parameters of the port-to-port RVL service curve are

$$R_{p2p}^{Fr} = \min [R_{SF} - r_{L_1}, R_{sch}], \quad T_{p2p}^{Fr} = \frac{R_{SF}T_{SF} + b_{L_1}}{R_{SF} - r_{L_1}} + T_{sch,con} \cdot \mathbf{1}_{\{r_{L_1} + r_T \geq R_{sch}\}} + \frac{l_{max}}{R_{sch}} . \quad (33)$$

5.1.3 Model Simulation

Figure 16 shows the test topology in which the comparison of theoretical and empirical latencies is shown. Table 3 summarises the parameters of the flows traversing the switch under test and Table 4 summarises the parameters of the switch under test. Figure 17 depicts the measured latencies and reconstructed latencies using the PBOO/PMOO approach.

Table 3: Flow parameters

Parameter	Value	Unit	Note
r_T	145	$kb \cdot s^{-1}$	limited by TestQoS
r_{L_1}	0 - 190	$Mb \cdot s^{-1}$	variable parameter
r_{L_2}	100	$Mb \cdot s^{-1}$	fixed parameter
b_T	8912	b	1024-byte long packet
b_{L_1}	12000/24000	b	2 IIs, 1500-byte long packets
b_{L_2}	12000	b	1 IIs, 1500-byte long packets
l_{max}^L	12000	b	1500-byte long packet

Table 4: Switch parameters

Parameter	Value	Unit	Note
R_{SF}	8000	$Mb \cdot s^{-1}$	
T_{SF}	90.5	μs	
R_{cla}	∞	$Mb \cdot s^{-1}$	
T_{cla}	0	μs	
R_{sch}	100	$Mb \cdot s^{-1}$	
T_{sch}	1.558	ms	$r_T + r_{L_1} \geq R_{sch}$

The following ad hoc approach for device concatenation is adopted: the left switch in Figure 16 is the device under test, the right switch is only used for deaggregation of the flows. The service curves for both switches are the same. Under the PBOO concatenation the maximum latency is given by

$$\hat{d}_{e2e}^{Fr}(F_{L_1}, F_{L_2}) = h \left(\alpha_T, \beta_{SW_1}^{Fr}(F_{L_1}, F_{L_2}) \otimes \beta_{SW_1}^{Fr}(F_{L_1}) \right) .$$

Both switches can be modelled by a service curve with parameters from (33). The first switch aggregates all flows in the SF and the aggregate of the F_T and F_{L_1} advance to the second switch through whose SF the flow deaggregate to continue to different OIs each, i.e., the OI of the SW2 is not congested. On that account, we can use the PMOO principle for the SF and OI of the first switch (SW1), and SF of the right switch (SW2). Having done so, we can append also the OI of SW2. Hence,

$$\beta_{SW_1}^{Fr} \otimes \beta_{SW_2}^{Fr} \approx [\beta_{SW_1,SF} \otimes \beta_{SW_1,OI} \otimes \beta_{SW_2,SF} - \alpha_{L_1}]^+ \otimes \beta_{SW_2,OI}. \quad (34)$$

It can be shown that the PMOO part in (34) yields in the same form as (33) only with the SF latency constant of $2T_{SF}$. Further, the OI of SW2 can be omitted. Hence, the end-to-end delay of the packets of the flow F_T is given by

$$\bar{d}_{e2e}^{Fr}(F_{L_1}, F_{L_2}) = \frac{2R_{SF}T_{SF} + b_{L_1}}{R_{SF} - r_{L_1}} + T_{sch,con} \cdot 1_{\{r_{L_1} + r_T \geq R_{sch}\}} + \frac{l_{max}}{R_{sch}} + \frac{b_T}{\min[R_{agg} - r_{L_1}, R_{sch}]}. \quad (35)$$

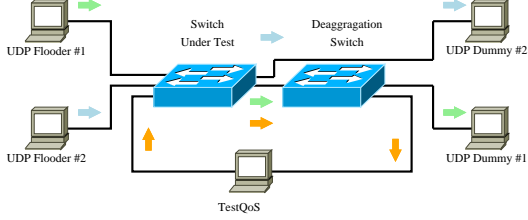


Figure 16: Switched topology for validation

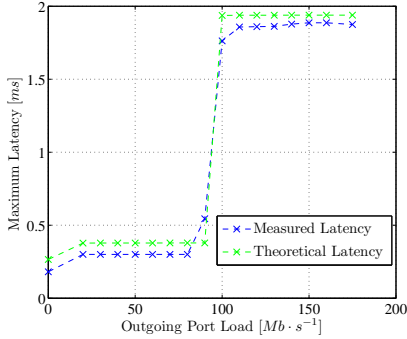


Figure 17: Reconstructed latency of packets passing two concatenated switches

The bounds seem pessimistic. On the other hand this approach is valid for concatenation of any number of switches and the theoretical latencies are comparable with the measured latencies introduced in test case SW.CON.

5.2 Cisco 2811 Router with HWIC-2FE Module

Cisco 2811 router houses two full-duplex ports with link capacity of $100 \text{ Mb} \cdot \text{s}^{-1}$. Using the HWIC-2FE module, the router is extended to house 4 fully routable ports. The router is a shared-memory router. Forwarding capacity can be increased by offloaded forwarding using CEF. However, due to shared resources, the router has to be considered as a router with a blocking SF, i.e., SF-B.

The OIs can be configured independently in terms of scheduling algorithms, buffer lengths, and others. The chosen scheduling mechanism for the validation is PQ. The number of PQ queues is four (see Figure 13). The queue assignment must be given by configuration. Only high-priority and low-priority queues are used for validation with 20 and 80 packet buffer, respectively.

Table 5: Reconstructed SF parameters

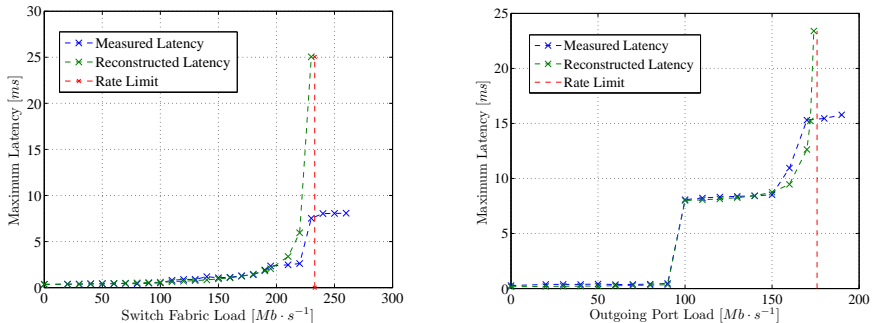
Scheduler	A [ms]	B [Mb · s ⁻¹]	(T _{OI} + T _{SF})' [μs]	R _{SF} Mb · s ⁻¹	T _{OI} + T _{SF} [μs]	d _{F_r} (0) - $\frac{b_T}{R_{OI}}$ [μs]
PQ	-78.098	-233.1	248.42	233.1	67.52	94.48
CBWFQ	-34.354	-199.4	75.00	199.4	75.00	94.93
WFQ	-129.423	-367.5	299.38	367.5	118.48	89.21

5.2.1 Switch Fabric Parametrisation

Firstly, the SF's parameters are identified. The values of the maximum latency measured by the TestQoS under different loads are shown in Figure 18 on the left. Identification according to Section 4.4 was carried out.

From the basic observation, it can be concluded that $T_{OI} \ll T_{SF}$. Consequently, (26) can be used. Using hyperbolic regression, the A and B parameters were inferred. The parameters R_{SF} and $T_{OI} + T_{SF}$ were obtained from (27). $b_T = 8192 b$ and $b_L = 11200 b$, which respects the packet lengths. The test topology of RTR.FL contained two more switches (not shown in the figure). It is possible to compensate for this difference by subtracting $180.9 \mu s$ which is the latency caused by two offloaded switches if the packet length is $1024 B$.

The obtained results are summarised in Table 5. The table shows also results of identification of the SF parameters under WFQ and CBWFQ scheduling mechanisms introduced in the dissertation. The fifth column shows the identified capacity R_{SF} . The fourth column of Table 5 represents the reconstructed values of $T_{OI} + T_{SF}$, and the sixth column represent the same values after compensation for the additional switches. The last column represents the direct value of $T_{OI} + T_{SF}$ inferred from the single latency value $\hat{d}_{F_r}(r_L = 0)$. The deviation of the reconstructed and single-point value is sufficiently small. The reconstructed dependencies are shown in Figure 18 on the left.


Figure 18: Reconstructed SF parameters (left) and OI parameters (right)

5.2.2 Outgoing Interface Parametrisation

The validation case is based on the PQ scheduling mechanisms at the OI. $R_{sch} = 100 Mb \cdot s^{-1}$ which is a limit imposed by the physical interface itself.

$T_{sch,con}$ must be inferred. From method description in the dissertation and Figure 18 value steps can be inferred. As the values for the OI identification are taken from the test case RTR.OPC, it holds that

$$T_{RTR,sch,con} = \hat{d}(r_L = 100 Mb \cdot s^{-1}) - \hat{d}(r_L = 90 Mb \cdot s^{-1}) = 8071.0 \mu s - 460.5 \mu s = 7610.5 \mu s. \quad (36)$$

Finally, using the approach introduced in the dissertation, the rate of the classifier was identified as $T_{cla} = 175.86 Mb \cdot s^{-1}$. Details are summarised in Table 6. The measured and reconstructed maximum latencies are shown in Figure 18 on the right. The important point is the vertical asymptote corresponding to T_{cla} .

Table 6: Reconstructed OI parameters

Scheduler	C [μs]	A [b]	B [$Mb \cdot s^{-1}$]	$2b_T + b_T$ [b]	R_{cla} [$Mb \cdot s^{-1}$]
PQ	108	-29418	-175.86	32192	175.86

5.2.3 Port-to-Port Service Curves

Figure 19 shows the topology chosen for the model validation. The topology consists of one switch from the previous validation case and the router configured with PQ scheduling mechanism at the OI and enabled CEF mechanism. The time-critical test flow $F_T(t)$ is disturbed by two additional flows; F_{L_1} passing along the same path to the same OI but with no priority indication and F_{L_2} passing over the SF and then continuing to a different OI. Hence, it is verified that the superposition of two loads can be handled by the model.

According to the PBOG/PMOO principle, the port-to-port service curve is given by (12) inferred for the SF-B/OI-PQ device type. The flows belong to the sets as follows:

$$\mathbb{S}_1 = \{\alpha_{L_1}^0, \alpha_{L_2}^0\}, \quad \mathbb{S}_2 = \{\alpha_{L_1}^1\}, \quad \mathbb{S}_3 = \emptyset. \quad (37)$$

Taking this distribution of additional flows into account and using Theorem 2.1, the port-to-port service curve will be calculated according to

$$\begin{aligned} \beta_{p2p}^{F_T} &= \beta_{SF}^{F_T} \otimes \beta_{OI}^{F_T} = R_{p2p}^{F_T} [t - T_{p2p}^{F_T}]^+, \\ \beta_{SF}^{F_T} &= [(\beta_{agg} \otimes \beta_{fwd}) - (\alpha_{L_1}^0 + \alpha_{L_2}^0)]^+, \quad \beta_{OI}^{F_T} = [\beta_{cla} - \alpha_{L_1}^1]^+ \otimes [\beta_{sch} - l_{max}]^+. \end{aligned} \quad (38)$$

Aggregation and forwarding can be merged in this case, thus service curve of SF can be inferred directly, subject that $\alpha_{L_1}^0 = r_{L_1}t + b_{L_1}$, and $\alpha_{L_2}^0 = r_{L_2}t + b_{L_2}$:

$$\beta_{SF}^{F_T} = (R_{SF} - r_{L_1} - r_{L_2}) \left[t - \frac{R_{SF}T_{SF} + b_{L_1} + b_{L_2}}{R_{SF} - r_{L_1} - r_{L_2}} \right]^+. \quad (39)$$

Subsequently, $\alpha_{L_1}^1$ must be calculated prior to inferring the service curve of the OI. It follows immediately from [10, 30] that

$$\alpha_{L_1}^1 = \beta_{SF}^{L_1} \otimes \alpha_{L_1}^0 = r_{L_1} + b_{L_1} + r_{L_1}T_{SF}^{L_1}. \quad (40)$$

Hence, it is necessary to compute also $\beta_{SF}^{F_{L_1}}$ to obtain $T_{SF}^{L_1}$.

$$\begin{aligned} \beta_{SF}^{F_{L_1}} &= [\beta_{SF} - (\alpha_T^0 + \alpha_{L_2}^0)]^+ = (R_{SF} - r_T - r_{L_2}) \left[t - \frac{R_{SF}T_{SF} + b_T + b_{L_2}}{R_{SF} - r_T - r_{L_2}} \right]^+ \Rightarrow \\ T_{SF}^{L_1} &= \frac{R_{SF}T_{SF} + b_T + b_{L_2}}{R_{SF} - r_T - r_{L_2}}. \end{aligned} \quad (41)$$

Consequently, $\alpha_{L_1}^1$ has the following form:

$$\begin{aligned} \alpha_{L_1}^1 &= r_{L_1}^1 t + b_{L_1}^1, \\ r_{L_1}^1 &= r_{L_1}, \quad b_{L_1}^1 = b_{L_1} + r_{L_1} \frac{R_{SF}T_{SF} + b_T + b_{L_2}}{R_{SF} - r_T - r_{L_2}}, \end{aligned} \quad (42)$$

Using Theorem 2.1, and the inferred parameters given in (42), the classifier has the service curve

$$\beta_{cla}^{F_T} = (R_{cla} - r_{L_1}) \left[t - \frac{R_{cla}T_{cla} + b_{L_1} + r_{L_1} \frac{R_{SF}T_{SF} + b_T + b_{L_2}}{R_{SF} - r_T - r_{L_2}}}{R_{cla} - r_{L_1}} \right]^+. \quad (43)$$

At last, the service curve of the scheduler must be inferred. The service curve is of the RVL type according to (5). Thus,

$$\beta^{F_T} = R_{sch} \left[t - T_{sch,con} \cdot \mathbb{1}_{\{r_{L_1} + r_T \geq R_{sch}\}} + \frac{l_{max}}{R_{sch}} \right]^+ . \quad (44)$$

Finally, by inserting (39), (43), and (44) into (38), the parameters of the port-to-port service curve are

$$\begin{aligned} R_{p2p}^{F_T} &= \min [R_{SF} - r_{L_1} - r_{L_2}, R_{cla} - r_{L_1}, R_{sch}], \\ T_{p2p}^{F_T} &= \frac{R_{SF} T_{SF} + b_{L_1} + b_{L_2}}{R_{SF} - r_{L_1} - r_{L_2}} + \frac{R_{cla} T_{cla} + b_{L_1} + r_{L_1} \frac{R_{SF} T_{SF} + b_T + b_{L_2}}{R_{SF} - r_T - r_{L_2}}}{R_{cla} - r_{L_1}} + T_{sch} \cdot \mathbb{1}_{\{r_{L_1} + r_T \geq R_{sch}\}} + \frac{l_{max}}{R_{sch}}. \end{aligned} \quad (45)$$

According to the EPBOO principle, the port-to-port service curve is given by (16) and (17) for the SF-B/OI-PQ architecture. The flows belong to the sets as follows:

$$\mathbb{S}_{R_1} = \{r_{L_1}, r_{L_2}\}, \mathbb{S}_{R_2} = \{r_{L_1}\}, \mathbb{S}_{R_3} = \emptyset, \mathbb{S}_{R_3'} = \{r'_{L_1}\}, \mathbb{S}_{B_1} = \{b_{L_2}\}, \mathbb{S}_{B_2} = \emptyset, \mathbb{S}_{B_3} = \{b_{L_1}\}. \quad (46)$$

Inserting this distribution into (16) and (17), and having evaluated the minima of the rate parameters, the port-to-port service curve's parameters are

$$\begin{aligned} R_{p2p}^{F_T} &= \min [R_{SF} - (r_{L_1} + r_{L_2}), R_{cla} - r_{L_1}, R_{sch}], \\ T_{p2p}^{F_T} &= T_{SF} + T_{sch,con} \cdot \mathbb{1}_{\{r_{L_1} + r_T \geq R_{sch}\}} + \frac{l_{max}}{R_{sch}} + \frac{b_{L_2}}{R_{SF}} + \frac{b_{L_1}}{R_{cla}} + \frac{b'_{L_1}}{R_{sch}}. \end{aligned} \quad (47)$$

5.2.4 Model Simulation

The validated topology incorporates OI congestion provided by the flow F_{L_1} and additional SF loading of the router provided by the flow F_{L_2} . The parameters of the router are introduced in Table 8. The parameters of the test flow are introduced in Table 7. The parameters of interest of the switch are introduced in Table 4.

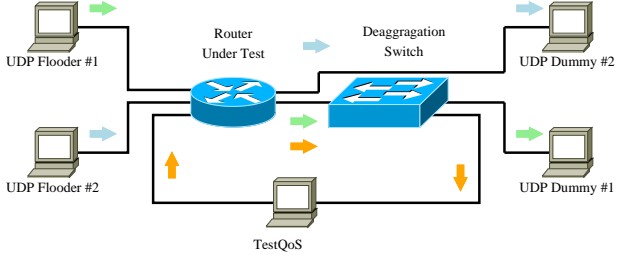


Figure 19: Routed topology for validation

Table 7: Flow parameters

Parameter	Value	Unit	Description
r_T	145	$kb \cdot s^{-1}$	limited by TestQoS
r_{L_1}	0 - 190	$Mb \cdot s^{-1}$	variable parameter
r_{L_2}	{0, 50, 100}	$Mb \cdot s^{-1}$	one value per test
b_T	8912	b	1024-byte long packet
b_{L_1}	24000	b	2 IIs, 1500-byte long packets
b_{L_2}	12000	b	1 II, 1500-byte long packets
l_{max}^I	12000	b	1500-byte long packet

So far, the port-to-port service curve of a device has been modelled according to the PBOO/PMOO approach. Nevertheless the end-to-end latency of a packet of a flow F_T is inferred using the TFA approach.

Table 8: Router parameters

Parameter	Value	Unit	Description
R_{SF}	233	$Mb \cdot s^{-1}$	capacity of the SF
T_{SF}	94	μs	
R_{cla}	175	$Mb \cdot s^{-1}$	
T_{cla}	0	μs	
R_{sch}	100	$Mb \cdot s^{-1}$	
T_{sch}	7.61	ms	

There are two reasons for doing so: (i) the computation simplifies significantly, and (ii) the latency caused by the switch is incomparably lower to the one of the router.

The resulting dependency shows the progress of the maximum latency of the flow F_T to the rate of the loading flow F_{L_1} and the flow F_{L_2} . The switch serves only for de-aggregation and thus behaves in a predictable way evident from Section 5.1. The maximum latency of the switch is approximately $\hat{d}_{p2p,sw}^{F_T} = 90.5 \mu s$ as observed in the previous validation case. The end-to-end delay of the packet from the flow F_T is according to the TFA given by the sum of the port-to-port latencies of the concatenated devices. Hence,

$$\hat{d}_{c2c}^{F_T}(F_{L_1}, F_{L_2}) = \hat{d}_{RTR}^{F_T}(F_{L_1}, F_{L_2}) + \hat{d}_{SW}^{F_T} = T_{RTR}^{F_T} + \frac{b_{F_T}}{R_{RTR}^{F_T}} + \hat{d}_{SW}^{F_T},$$

where the parameters are those introduced in (45) and the values are introduced in Table 7 and Table 8. Figure 20 on the left shows the measured and theoretical latencies for the mixed topology and the given load distribution.

The inferred parameters based on which the theoretical latencies were inferred represent well enough the upper bound to the measured latencies. In the case with SF load of 0 and 50 $Mb \cdot s^{-1}$, the latency growth to saturation is given by the load approaching the classifier maximum capacity. In the case with SF load of 100 $Mb \cdot s^{-1}$, the latency growth to saturation is given by the load approaching the SF capacity.

The model stops to be valid once the limit is reached. In such a case, some of the traffic starts to be dropped and the saturation in latency appears.

Figure 20 on the right shows the measured and reconstructed latencies using the EPBOO approach for the mixed topology and the given load distribution. It can be observed that the EPBOO approach does not represent the real behaviour satisfyingly.

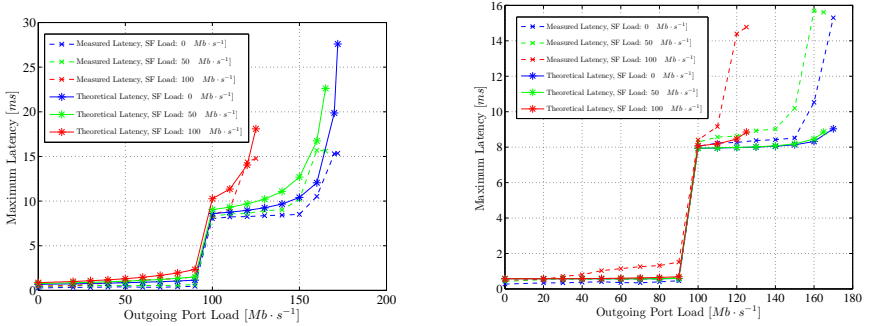


Figure 20: Reconstructed latency of packets passing a router with compound loading. PBOO on the left and EPBOO on the right

6 ASSESSMENT AND CONCLUSIONS

6.1 Validation Assessment

In the first validation case, behaviour of two overloaded concatenated switches was modelled. The step-wise change of the latency after crossing values can have two reasons. Yet, even combination of both. The first reason may be execution of the QoS mechanisms which consume more resources. The second reason may be that as the additional flows's rate approaches the OI's capacity, the latency asymptotically increases to infinity. However, as the interface cannot transmit flow higher than the interface capacity allows and the buffer is finite, the latency saturates. Nevertheless, both reasons can be with satisfying precision modelled by the RVL function as is shown by these validations.

Generally, the behaviour of the HP 1800-8G Procurve switch is well representable by a simple model as can be seen in Figure 16.

In the second validation case, behaviour of a topology accommodating one router and one switch was modelled. The most distinguishing factor is variable contribution of the SF to the port-to-port packet latency. SF parameters were firstly identified by the procedure described in Section 4.4 and using the measurements obtained in RTR.FL test.

Furthermore, parameters of the OI were identified from the RTR.OPC test. The classifier and scheduler were jointly modelled by the RVL service curve. In this stage, the simulation result is very precise.

It can be seen that the PBOO approach represents well the upper-bounds of latency of the traversing packet. Contrary to this, EPBOO approach does not fit the real behaviour at all. Most probably, it is not possible to account for multiplexing only once because the packets arbitrate at more points of the path.

6.2 Conclusions

It was shown by initial experiments that the QoS-enabled router decreases packet latency and jitter. However, should such mechanisms be employed in industrial safety-relevant environment, worst-case temporal analysis must be performed in order to provide reasonable evidence arguing for the IP-based communication applicability in industrial environment. The problem area is twofold: (i) qualitative analysis is required to reveal dominant factors influencing the QoS behaviour of the IP-based networking devices, and (ii) quantitative analysis must be performed to reveal upper bounds of latencies, jitters, bandwidth, and packet loss.

Qualitative analysis was subject to structured empirical observations and extensive studies into networking device architectures. Results of the qualitative analysis were used for reasoning about the model structure of a networking device. Quantitative analysis was subject to dedicated empirical observations with focus on measurement stability and result reliability. Results of the quantitative analysis were used for identification of the parameters of the networking device and final validation of the device models.

Empirical observations introduced in Section 3 were performed by TestQoS test bed which was developed by the author within the Virtual Automation Networks (VAN) project dealing with the issues of the IP-based industrial communication. TestQoS measures latencies of packet injected into the network/device under test. The salient feature of the TestQoS is the precision of the measured packet latency which reaches the resolution of 10 ns. Drawback of the TestQoS is the limited rate of the packet injection which reaches ca $150 \text{ kb} \cdot \text{s}^{-1}$.

Having investigated several approaches to temporal analysis of IP-based communication networks, application of the network calculus for network/device modelling proved to be very relevant on one hand, and very challenging due to its practical immaturity on the other hand. According to the available resources, such a massive practical application of network calculus has never been performed so far.

There were many obstacles to be overcome on the way to consistent networking device models. For instance, bi-modal behaviour of latency depending on the device load, insufficient closed-form representations of lossy nodes, lack of parameter identification methods. Most of these challenges were tackled by developing the needed network calculus extensions summarised in Section 4.1. Four exemplary models were inferred within the dissertation as a result of classification of different networking device architectures and empirically identified dominant factors; two of them introduced in Section 4. These types form a basis for a future broader taxonomy of designs. Special attention has been paid to reusability of the model when employing different networking devices than those considered in this work. Approaches to model design of SFs and OIs have been proposed. Two concatenation approach were considered for the design modelling: PBOO/PMOO and EPBOO.

Final validation of the models of HP 1800-8G ProCurve switch and Cisco 2811 router is introduced in Section ???. The proposed method for model parametrisation proved applicable. PBOO/PMOO approach provided good results and the worst-case upper bounds of latency conform the measurements. EPBOO approach failed to deliver the required behaviour.

6.3 Unresolved Issues and Further Research

Despite extensive research effort related to this topic, there are numerous issues which require further effort. The following points suggest the successive research agenda in this field.

Extension of the TestQoS capabilities. The most limiting factor is insufficient rate of the generated test traffic to be injected into the network under test. Upgrade of the TestQoS is subject to significant change in the architecture towards multi-threaded version of the traffic generation block. Accomplishment of this step would allow for generating of self-loading traffic necessary for more advanced analysis.

Validation of the model in a broader scope. Accomplishment of the previous goal would allow for investigation of new test scenarios. For instance, full investigation of the WRR, WFQ, and CBWFQ scheduling mechanisms, arbitration of more high-priority flows, etc. Consequently, modelling approach introduced in the dissertation could be verified and the model validity could be extended accordingly.

Service curves of other scheduling mechanisms. The scheduling mechanisms regarded in the previous point could be represented more rigorously and full verification should be performed. This is only feasible subject to extension of the TestQoS application or if commercially available tools are employed. Promising and cost-effective validation of the models could be performed in a network on chip (NoC) application.

Representation of loss nodes. Network calculus framework has a high application potential due to its modularity, i.e., block-based nature. Despite this fact, there is no closed-form service curve representing a *loss node*. Establishment of a service curve with losses would extend validity of the models to boundary conditions, which are of particular interest for safety-critical applications.

Analysis of complex topologies. Despite the fact that applicable models of networking devices have been inferred, the scope of the research is more ambitious. A formalised approach to concatenation of numerous networking devices should be established. It has already been proven that EPBOO approach has not provided satisfying results in this case, hence TFA or SFA concatenation approach should be considered. A proper trade-off between the complex network model and bound tightness has to be found.

Promotion of the modelling framework to an engineering tool. Tool development requires a formalised modular approach with clearly defined interfaces. This issue has been paid proper attention throughout the dissertation, especially in case of the typology definition of the models of networking devices. Similar approach has already been applied in case of *ns2* framework, which is however dedicated rather to theoretical blocks without device oriented semantics. On the other hand, the intended approach could potentially deliver support for assisted model structure definition, similar to Opnet modelling environment.

6.4 Final Remarks

The presented dissertation establishes a cornerstone for the upcoming research in worst-case temporal performance analysis based on network calculus and related SW engineering challenges in order to successfully accomplish its mission. Motivation to this research topic was formulated based on industrial innovation requirements presented by researchers in the VAN project. The results should be applied *ibidem*, i.e., industrial automation. Yet, performance analysis, a topical superset of the temporal performance analysis, has been gaining importance with the growing market of the embedded systems which are recently undergoing an evolutionary leap towards systems of systems (SoS) where networking and its temporal performance are the main non-functional aspects.

References

- [1] ALMQUIST, P. *Type of Service in the Internet Protocol Suite (RFC 1349)*. 1992 Updated by RFC 2474.
- [2] BERAN, J., ELIA, A., HUNDT, L., HEUTGER, H., MEO, F., MESSERSCHMITT, R., SCHÖNMÜLLER, B., WERNER, T., WOLFRAMM, M. D04.2.1 - results of modelling of rt mechanisms in automation systems and rt extensions of existing industrial solutions. Deliverable, VAN Project, 2006.
- [3] BERAN, J. Virtual automation networks - challenge in industrial automation. In *Proceedings of the IFAC Workshop on Programmable Devices and Embedded Systems. PDES 2006*, p. 473–478, 2006.
- [4] BERAN, J., ZEULKA, F., FIEDLER, P. Findings on qos metrics of I3 network devices intended for future factory automation. In *Proceedings of the IFAC Workshop on Programmable Devices and Embedded Systems. PDES 2009*, p. 208 – 213, 2009.
- [5] BERAN, J., FIEDLER, P., ZEULKA, F. Virtual automation networks: An evolutionary step towards industrial internet. *IEEE Industrial Electronics Magazine*, vol. 4, no. 3, p. 20, September 2010.
- [6] BERAN, J., ZEULKA, F. Evaluation of real-time behaviour in virtual automation networks. In *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, 2008.
- [7] BERAN, J., FIEDLER, P., ZEULKA, F. Rate-variable-latency service curve as an extension to network calculus. In *Control and Automation, 2009. MED '09. 17th Mediterranean Conference on*, p. 286–291, June 2009.
- [8] BERAN, J., FIEDLER, P., ZEULKA, F. Modeling a router temporal performance using network calculus. In *Proceedings of the 8th International PhD Student's Workshop on Control and Information Technology*, p. 197–205, 2009.
- [9] BLACK, S., BLACK, D., CARLSON, M., DAVIES, E., WANG, Z., WEISS, W. *An Architecture for Differentiated Services (RFC 2475)*. 1997 Updated by RFC 3260.
- [10] BOUDEC, J.-Y. L., THIRAN, P. *Network Calculus - A Theory of Deterministic Queuing Systems for the Internet.*, vol. 2050 Springer Verlag, 2001.
- [11] CHAO, H. J., LIU, B. *High Performance Switches and Routers*. Wiley-IEEE Press, 2007.
- [12] CHERTOV, R.; FAHMY, S. S. N. B. A black-box router profiler. 05 2007.
- [13] COLLECTIVE. *IAONA Handbook - Industrial Ethernet*. IAONA e.V, 2005.
- [14] CRUZ, R. L. A calculus for network delay: Part ii: Network analysis. In *IEEE Transactions on Information Theory*, vol. 37, p. 132–141, 1991.
- [15] DOOLEY, K., BROWN, I. J. *Cisco IOS Cookbook*. O'Reilly, 2003.
- [16] FARIA, F. D., STRUM, M., CHAU, W. J. A system-level performance evaluation methodology for network processors based on network calculus analytical modeling. *VLSI, IEEE Computer Society Annual Symposium on*, vol. 0, p. 265–272, 2007.
- [17] FIROIU., V., BOUDEC, J.-Y. L., TOWSLEY, D., ZHANG, Z.-L. Theories and models for internet quality of service. In *Proceedings of the IEEE*, 2002.
- [18] FORD, M. *Internetworking Technologies Handbook*. Cisco Press, 1998.
- [19] GEORGES, J.-P., DIVOUX, T., RONDEAU, E. Validation of the network calculus approach for the performance evaluation of switched ethernet-based industrial communication. *Proceedings of the 16th IFAC World Congress*, 2005.
- [20] GEORGES, J.-P., DIVOUX, T., RONDEAU, E. Strict priority versus weighted fair queuing in switched ethernet networks for time-critical applications. In *Proceedings of the 19th IEEE International and Distributed Processing Symposium*, 2005.
- [21] GEORGES, J.-P., DIVOUX, T., RONDEAU, E. Comparison of switched ethernet architectures models. In *Proceedings of IEEE Conference on Emerging Technologies and Factory Automation 2003*, 2003.

- [22] GEORGES, J.-P., DIVOUX, T., RONDEAU, E. Confronting the the performances of a switched ethernet network with industrial constraints by using network calculus. *International Journal of Communication Systems*, vol. 18, p. 877–903, 2005.
- [23] GEORGES, J.-P., DIVOUX, T., RONDEAU, E. A formal method to guarantee a deterministic behaviour of switched ethernet networks for time-critical applications. In *Proceedings of the IEEE International Symposium on Computer Aided Control Systems Design*, p. 255–260, 2004.
- [24] GROSSMAN, D. *New Terminology and Clarifications for DiffServ (RFC 3260)*. 2002.
- [25] GUPTA, R. A., CHOW, M.-Y. Performance assessment and compensation for secure networked control systems. In *Proceedings of the 34th Annual Conference of IEEE Industrial Electronics (IECON 2008)*, p. 2929–2934, 2008.
- [26] HANZALEK, Z., PACHA, T. Use of the fieldbus systems in academic setting. In *Proceedings of Real-Time Systems Education III*, p. 93–97, 1998.
- [27] JASPERNEITE, J., NEUMANN, P. Deterministic real-time communication with switched Ethernet. In *Proceedings of 4th IEEE International Workshop on Factory Communication Systems*, p. 11–18, 2002.
- [28] JASPERNEITE, J., NEUMANN, P. Measurement, analysis and modelling of real-time source data traffic in factory communication systems. In *Proceedings of IEEE International Workshop on Factory Communication Systems*, p. 327–333, 2000.
- [29] PARK, S. G. Fieldbus in iec61158 standard. In *Proceedings on the 15th CISL Winter Workshop Kushu*, 2002.
- [30] PONCE, V., M. *Engineering Hydrology: Principles and Practices*. Prentice Hall, 1994.
- [31] SCHMITT, J. B. Network calculus: what it can do for you and where it needs your help. In *AINTEC '08: Proceedings of the 4th Asian Conference on Internet Engineering*, p. 57–58, New York, NY, USA, 2008. ACM.
- [32] SCHMITT, J. B., ZDARSKY, F. A., FIDLER, M. Delay bounds under arbitrary multiplexing: When network calculus leaves us in the lurch.... In *Proceedings of the 27th IEEE Conference on Computer Communications*, p. 2342–2350, 2008.
- [33] SCHMITT, J. B., ZDARSKY, F. A. The disco network calculator: a toolbox for worst case analysis. In *Proceedings of the 1st international conference on Performance evaluation methodologies and tools*, 2006.
- [34] ZAMPIERI, S. Trends in networked control systems. In *Proceeding of the 17th IFAC World Congress, Seoul, Korea*, 2008.

Curriculum Vitae

Jméno:	Jan Beran, Ing.
Narozen:	14. 10. 1980, Vyškov
E-mail:	beranj@feec.vutbr.cz
Adresa:	Špačkova 3876/8 Brno, 63600 Česká republika

Vzdělání

1992-2000	Gymnázium Zlín, Lesní čtvrt', Zlín, CZ,
2000-2005	Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, <i>obor: Kybernetika, automatizace a měření</i>
2004-2004	zahraniční pobyt na Tampere University of Technology, Tampere, Finsko
2006-2010	Doktorské studium na Vysokém učení technickém v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, <i>obor: Kybernetika, automatizace a měření - kombinovaná forma</i>

Projekty, praxe

2004-2006	Lokalizace automatizačního software firmy Siemens
2005-2006	Spolupráce s firmou BetaControl, <i>projekt: Systém pro monitorování teploty hald</i>
2005-2009	Virtual Automation Networks (VAN), 6thFP, IST
2010	Honeywell International, s.r.o., Advanced Technology Europe - Platform Systems Group, <i>pozice: R&D Scientist</i>

Publikace

- [1] BERAN, J. Virtual automation networks - challenge in industrial automation. In *Proceedings of the IFAC Workshop on Programmable Devices and Embedded Systems. PDES 2006*, p. 473-478, 2006.
- [2] BERAN, J., ZEŽULKA, F. Evaluation of real-time behaviour in virtual automation networks. In *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, 2008.
- [3] BERAN, J., FIEDLER, P., ZEŽULKA, F. Modeling a router temporal performance using network calculus. In *Proceedings of the 8th International PhD Student's Workshop on Control and Information Technology*, p. 197-205, 2009.
- [4] BERAN, J., FIEDLER, P., ZEŽULKA, F. Rate-variable-latency service curve as an extension to network calculus. In *Control and Automation, 2009. MED '09. 17th Mediterranean Conference on*, p. 286-291, June 2009.
- [5] BERAN, J., ZEŽULKA, F., FIEDLER, P. Findings on qos metrics of l3 network devices intended for future factory automation. In *Proceedings of the IFAC Workshop on Programmable Devices and Embedded Systems. PDES 2009*, p. 208 - 213, 2009.
- [6] BERAN, J., FIEDLER, P., ZEŽULKA, F. Virtual automation networks: An evolutionary step towards industrial internet. *IEEE Industrial Electronics Magazine*, vol. 4, no. 3, p. 20, September 2010.