Dual Mathematical Model for Calculating of Deep Packet Inspection

Boris Goldstein, Vadim Fitsov
Bonch-Bruevich Saint-Petersburg State University of Telecommunications
Saint-Petersburg, Russia
bgold@niits.ru, vf@sotsbi.ru

***Abstract*—This article describes a dual mathematical model for calculating deep packet inspection systems, based on the Norros model and the Ventcel-Ovcharov model. The modified Norros model is used to calculate the hardware filter for the deep packet inspection system. The Ventcel-Ovcharov model is used to calculate the remaining specialized servers of the deep packet inspection system. The created simulation model of the deep packet inspection system in GPSS World is described. The results of the mathematical and simulation modeling are compared.**

I. Introduction

Deep packet inspection (DPI) systems are used in the modern world to control and offload their communication networks, behavioral targeting, and enforce the laws of the country. However, when telecommunications operators or corporations buy expensive DPI equipment, they do not know how much performance is required for such a system in their particular case.

In the absence of a mathematical model, the choice of the DPI system performance is based on the bandwidth of the channels. Mathematical calculation would allow to more accurately determine the required DPI system performance in each specific case.

This paper describes a dual mathematical model for calculating of the DPI system. DPI mathematical and simulation models would reduce the purchase cost of DPI systems and to avoid overloads. The mathematical and simulation models can determine the parameters of the DPI architecture and increase effectiveness of DPI system.

There are various mathematical models, which sould be applied on the basis of features of packet traffic coming in the system. Modern western research suggests that network traffic is similar to itself or fractal in structure (pulsating on a wide time scale). This kind of traffic is most successfully described by the Pareto and Weibull distributions than by other distributions. In [1], the successful application of fractal Brownian motion (FBM) for the mathematical description of self-similar flows with a long range dependence (LRD) is described. For servers receiving packet traffic, you should use the G/G/V model, but there is currently no solution for it [1]. In addition, there is an alternative approach to calculating the load on equipment presented by Ikka Norros [2], [3] and described in the works of Murali Krishna. P, Vikram M. Gadre, Uday B. Desai [4], Christian Grimm, Georg Schluchtermann [1] (hereinafter Norros model).

When there are several interacting queuing systems (QS), they make up a queuing network (QN). In a QN, the interest is the parameters of the output after processing in QS1, which determine the models that can be used to describe the subsequent QS (QS2). Proceeding from the requirement to avoid packet loss, the QN (consisting of QS1 and QS2) is designed in such a way that it can be represented as a system with an infinite queue, which is important for the cases considered below.

According to Burke's theorem for QS1 (M/M/V and M/M/1), the distribution of time intervals between outgoing requests, as well as the time intervals between incoming requests, are distributed exponentially with the same parameter.

In [5], it is said that for a primary server G/G/1 with an unlimited capacity storage unit operating without overloads, the intensity of the outgoing flow of requests is equal to the intensity of the incoming flow (since the mathematical expectations of the intervals between successive requests at the exit and the entrance coincide). According to the study [6], the value of the variation coefficient of the resulting flow of requests (packet traffic) on the input of QS1 of a large number of sources with similar distributions for certain cases approaches one. The subsequent verification carried out in [6] using the Kolmogorov-Smirnov criterion showed that the addition of a large number (more than 100) of flows with the Weibull distribution gives the resulting flow a manner similar to Poisson flow.

Let us consider the cases in which a forced assumption will be made about the exponential distribution of the flow exiting QS1 and entering the QS2, in order to compare the results of the QS2 calculations.

In the first case, QS1 receives aggregated packet traffic received from more than 100 sources, where each traffic from each source can be approximately described by the Weibull distribution. Then, according to [6], we assume that the distribution of the flow of requests entering the QS1 may be close to the Poisson distribution. Then, according to [5], let us assume for one QS1 service device that the output flow entering QS2 is also close to the Poisson distribution.

In the second case, if we assume that QS1 processes requests according to an exponential distribution, and is in a mode when the average intensity of requests is equal to the average intensity of request processing, but the system remains in a stable state and the waiting time in the queue does not become infinite. Then, according to Burke's theorem, the intensity of the output flow of requests from QS1 to QS2 will be distributed in the same way as the service time in QS1. This means that if we assume that requests in QS1 are treated exponentially, then the output flow will have an exponential distribution.

The third case according to the conditions and assumptions of the first case, it is assumed that a flow close to Poisson arrives at the QS1. QS1 must process requests according to an exponential distribution. Then, according to Burke's theorem, the distribution of time intervals between outgoing requests, as well as the time intervals between incoming requests, are distributed exponentially. In this case, there is no limitation per 1 device as in the first case, and there is no limitation of the mode necessarily operating in a limited but stable state, as in the second case.

Further mathematical model of the QS2 is carried out based on these three cases.

For QS2, it is possible to use the models with an infinite queue M/M/V and M/G/V described by Cox, processor sharing (M/M/V/PS) [7]–[9] or so-called Ventcel-Ovcharov model with an equal mutual assistance [10]–[13] (where several devices work to serve one request).

The processor sharing (PS) model, described by Kleinrock in 1967 [7] is widely known. PS or EPS (egalitarian processor sharing) is a service policy in which all requests are served simultaneously. Each newly arrived request receives an equal share of the bandwidth. This does not imply the presence of a queue. Significant results in the systems with fair processor sharing, including the solution of the previously insoluble task finding the stationary distribution of the request stay time in the system, research were introduced in theory by S.F. Yashkov [8], [9]. ESP is very close to the so-called Ventcel-Ovcharov model with full mutual assistance.

Since the DPI system must process all requests, to simplify the calculations of the QS2 mathematical model, it is advisable to take a so-called Ventcel-Ovcharov model with an infinite queue and with equal mutual assistance [10] mentioned, but not completely described, in the works of Ventcel.

Thus, for the mathematical description of the servers of the DPI network architecture, the approach of using two mathematical models can be used. In other words, a dual mathematical model.

II. DPI AND RELATED WORK

In 2012, the International Telecommunication Union (ITU) officially approved the Y.2770 Requirements for deep packet inspection in Next Generation Networks standard [14]. This recommendation defines the requirements for deep packet inspection. DPI - is a analysis, according to the layered protocol architecture OSI, of payload and/or packet properties deeper than protocol layer 2, 3 or 4 header information, and other packet properties in order to identify the application unambiguously [14]. Flow - is a set of IP packets, witch have a set of common properties. A flow (associated with a specific user application or service protocol) is usually identified using address information from layers 2-4 of the OSI model. So DPI analyzes the first packets of a traffic flow or all packets passing through it. Mirrored traffic is analyzed to avoid DPI impacts on the QoS of passing traffic. Or the DPI system itself passes traffic until the analysis is completed, processing it according to the system's default policy. To indicate the time of analysis of a new unknown flow on DPI servers and apply the appropriate policy to it, the concept of delay in corrective actions was introduced.

The basis of the DPI system is the Bypass server, the hardware filter (HWF: DPI scanner and DPI action execution function), DPI analyzer (DPI-An or Front-End), PCRF (Policy and Charging Rules Function) and Back-End at Fig. 1. Each of the DPI servers performs its own tasks and actively interacts with the rest.

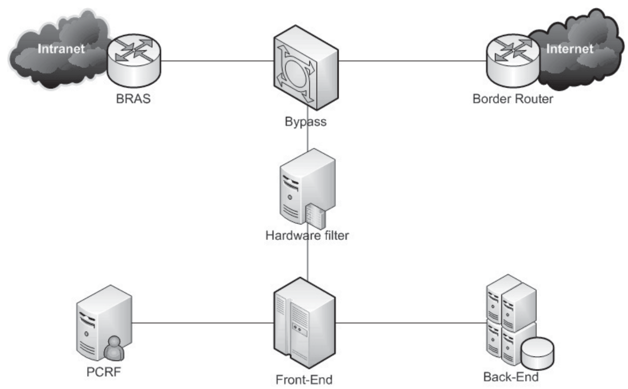

Fig. 1. Deep packet inspection system architecture

Bypass is necessary in the event of a DPI system failure, in which it transfers the traffic flowing through it to the system's hardware filter directly to the network without analysis, but also without significant losses or delays. A hardware filter performs basic traffic processing functions: policy enforcement (block or pass), statistics collection, bandwidth control.

Front-End (Analyzer) - is the main element of the system, as it analyzes traffic flows. In addition, line board is used to receive and transmit traffic. Having recognized the application that generated the traffic flow, Front-End asks the PCRF server for a decision on what to do with this traffic. Further, based on this decision, it receives more detailed instructions on filtering from the Back-End server. Then it gives the flow and instructions for execution on the hardware filter.

Back-End - is a repository of information (the policies rules, signatures and others). PCRF is a server for real-time policy management and decision.

In [15] for the calculation of the communication network Levakov A.K. represents it as a queuing network graph, where each node is a queuing system (QS) with waiting. Each route is a multiphase queuing system, the study of which is a complex process. This allows for the simplest cases to obtain the probability distribution function, the probability of blocking a service at a certain point in the network, the average load of a network link. To perform calculations, mathematical methods with software packages are used.

Based on the generalized DPI architecture (Fig. 1), the DPI mathematical model can be represented as queuing network. The processing algorithms used on the Front-End, PCRF and Back-End servers have a different effect on system performance. Given the insignificant load on some of the servers, one can limit ourselves to two servers: QS1 (hardware filter together with Bypass) with incoming packet traffic and QS2 (Front-End) with a flow of requests from QS1. From the previously given list of suitable models for the dual mathematical model, the Norros model and the Ventcel-Ovcharov model were selected. The primary flow of requests goes to the hardware filter, some of them generate a new flow of requests to the Front-End, which in turn instructs the hardware filter how to process unknown flows. The hardware filter then applies the necessary policies to the newly identified traffic flows. In this case, the average delay of corrective actions of the DPI system will be the sum of the time that requests are in queues and in processing on DPI servers.

III. The dual mathematical model

A. Formalization DPI input parameters

For the practical use of mathematical models, it is necessary to determine methods for obtaining quantitative characteristics of the operating conditions of the DPI system and formalize some features of the operation of the system itself.

To calculate, you need to know the intensity of incoming requests for the DPI system. That depends on the intensity of incoming packets to the hardware filter (λ_0). Therefore, it is rational to use the statistics of the transmitted traffic on the network where you plan to install the DPI system. A peculiarity of DPI is that QS1 (hardware filter) processes all incoming packets and identifies traffic flows from them, and QS2 (Front-End) receives a request to analyze a specific flow. For analysis few packets of the flow (n_f) is transmitted, about which studies have been carried out [16], [17]. Only new unknown flows are analyzed, already known flows are processed according to the previously set rules and are not sent to the Front-End.

From the traffic statistics collected by wireshark or cisco NetFlow, you can get the number of packets and the number of flows during statistical analysis, and then the average number of packets in the flow (n_{af}) and the probability that the flow was previously known (or unknown). Usually the probability that the flow was previously known (P_{kn}) is 70-90%.

The peculiarities of interaction between specialized DPI servers determine the number of requests arising on each of them, and hence the requirements for their performance. To determine the total number of requests coming to the DPI system hardware filter, one should take into account its operating mode, specified by the coefficient - S.

If, before the analysis of the unknown flow is completed, subsequent packets are discarded, then the value of the coefficient $S = -1$ should be used. If they are transmitted further along the network, according to the default policy, then $S = 0$. And when they are buffered in anticipation of the analysis result, and then transmitted, then $S = 1$. At the end of formula (1) and before multiplying by the intensity of the incoming packets by the hardware filter (λ_0), the added "1" means the work spent on reading the headers of all incoming packets at the first four layers of the OSI model. Thus, the number of requests that need to be processed by the hardware filter, including interaction with the Front-End, is calculated using the formula (1).

$$\lambda_{hw} = \left[P_{kn} + (1 - P_{kn}) \times (\frac{n_f + 1}{n_{af}} + S) + 1\right] \times \lambda_0 \quad (1)$$

As can be seen from (2), the number of requests to the Front-End depends on the intensity of arrival of packets to the hardware filter (λ_0), the probability of an unknown flow ($1 - P_{kn}$), and the number of analyzed packets (n_f) per flow. The number of requests coming from hardware filter to Front-End (2).

$$\lambda_{fe} = (1 - P_{kn}) \times (\frac{n_f + 1}{n_{af}}) \times \lambda_0 \quad (2)$$

So, the traffic analysis server (Front-End) processes the first few packets of the new flow (n_f) detected by the hardware filter as a single request. Front-End determines the required policy and passes the directions back to the hardware filter. In some cases, the traffic analysis server requests the policy from the PCRF decision server to apply it. If necessary, variables can be easily added to the formula (2) to account for interaction with the PCRF and Back-End servers (Fig. 1). This article further does not include requests sent to or received from PCRF and Back-End servers, due to the fact that the number of such requests is relatively small and has low hardware resource requirements.

The parameters defined here are conventionally divided into two groups. The first group of parameters is based on statistics collected from the communication network and allows you to describe the transmitted traffic. It includes some parameters related to the Norros model, some of which are defined in Table I. The second group of parameters characterizes the DPI system (S - operating mode and V - the number of devices for each server). Both groups of parameters are presented in Table II.

B. Elaboration of the Norros model and hardware filter

An alternative to the classical approach for calculating a system with incoming packet traffic is the approach based on the Iikka Norros model [2], [3] where fractional Brownian motion (FBM) is used to describe incoming packet traffic.

To understand it, it will be easier to start by comparing the stability condition of the system for the Norros model (QS1) (4) and the classical QS model (QS2) (3). If we compare the classical model of describing the processing of requests with the Norros model, then we can put in parallel the average value of incoming traffic (m) with the intensity of arrival of requests (λ), and the bandwidth of the system (C) with the intensity of service of requests (μ). Their ratio in both models sets the stability condition for the system ($\rho < 1$).

In the classic model:

$$\rho = \frac{\lambda}{\mu} \quad (3)$$

In the Norros model:

$$\rho = \frac{m}{C} \quad (4)$$

where m is the average amount of incoming traffic, and C is the system bandwidth.

The lower the system bandwidth, the more LRD arises. In [1], the formulas for the probability of waiting in the queue are given, which for the Hurst parameter (H) is greater than 0.5, repeat the Weibull distribution formulas. Formulas for the number of requests in the system and the received traffic for a given period are also described in [1]. The probability of waiting for a request in the queue, represented in the Norros model by formula (5), is then used to construct a mathematical model of the DPI system.

$$P(X_t > x) \approx \exp(-\frac{(C-m)^{2\times H}}{2\times\varphi(H)^2\times a\times m}\times x^{2-2\times H}) \quad (5)$$

where the parameter a is the characteristic moment for the FBM distribution, the $\varphi(H)$ coefficient is defined in (6), and the number of requests in service (x) are presented in Tables I and II.

$$\varphi(H) = H^H \times (1-H)^{1-H} \quad (6)$$

TABLE I. VALUES OF THE HURST PARAMETER AS A PROPERTY OF SELF-SIMILARITY AND CHARACTERISTIC MOMENT OF FRACTIONAL BROWNIAN MOTION FOR DIFFERENT TYPES OF TRAFFIC

HURST PARAMETER	TYPE OF TRAFFIC
$0 < H < 0.5$	zig-zag Brownian motion (non-renewable process, non-self-similar)
$H = 0.5$	chaotic movement (Markov flow, SRD (short range dependence))
$0.5 < H < 1$	renewable process (self-similar process); some of the formulas repeat the formulas for the Weibull distribution
characteristic FBM moment	specifies the moments, the shape of the additional distribution (tail), asymptotic behavior
$a = 1$	Cauchy distribution
$a < 2$	non-symmetric distribution
$a = 2$	Gaussian (normal) distribution with zero mean

To obtain the average time spent on processing a request in QS1 (T_{hw}) using the Norros mathematical model (7), it follows using the formulas for the probability of a queue ($P(X_t > x)$ determined by formula (5)) and the number requests in service (x), it is necessary to obtain the length of the queue, the number of requests in the system, the time spent by requests in the system [18]. Additionally, the number of servicing devices (V_{hw}) in QS1 must be taken into account.

The queue length for hardware filter, among other things, depends on the number of servicing devices (V_{hw}) - in HWF, parameter ρ (4) and can be obtained based on the probability of waiting for a request in the queue on HWF (5).

$$T_{hw} \approx \frac{\rho\times\exp(-\frac{(C-m)^{2\times H}}{2\times\varphi(H)^2\times a\times m}\times x^{2-2\times H})}{(V_{hw}-\rho)\times m} + \frac{x}{m} \quad (7)$$

Moreover, ρ is determined by formula (4), m by formula (1), and the coefficient $\varphi(H)$ by formula (6). Thus, it becomes possible to calculate the time spent by the request in the system using the Norros model.

C. Ventcel-Ovcharov model and Front-End

Previously, a forced assumption was made in the cases considered in which the exponential distribution of the flow leaving the hardware filter and entering the Front-End. That allows to use simpler classical models.

However, modern service systems usually use a central processor for various tasks, as opposed to highly specialized devices serving each one task sequentially. And the architecture with virtualization implies the allocation of the necessary computing power to the virtual server as needed within the specified limits. The Ventcel-Ovcharov model and Processor Sharing models are well suited for use not only for processor and multiprocessor systems, but also for servers in virtualization systems. In particular for the case of using a virtualized DPI system.

The Egalitarian Processor Sharing (EPS) model is very close to the so-called Ventcel-Ovcharov model with full mutual assistance. QS with mutual assistance is a QS in which several devices work to serve one request. Mutual assistance models for simplification in this article are called the Ventcel-Ovcharov model, since they were described in their writings [11]–[13]. Ventcel and Ovcharov distinguish two types of mutual assistance: full and partial (equal). This article discusses equal mutual assistance. Since the DPI system must process all requests, to simplify the calculations of the Front-End mathematical model, it is advisable to take a model with an infinite queue and with equal mutual assistance [10] mentioned, but not completely described, in the works of Ventcel.

The concept of the model with equal mutual assistance is to combine channels into groups for the joint service of requests. In this case, a system with equal mutual assistance will have 3 modes of operation, shown in Fig. 2: I - the number of requests is less than the maximum number of groups (like a classical QS), II - the number of requests is greater than the maximum number of groups, but less than the number of channels (transient mode), III - the number of requests is greater than the number of channels (like a classical QS). One of the advantages of the considered mathematical model is the use of all possible resources of the system before the number of requests equals the number of channels.

Mode I of operation implies the formation of channel groups. In this mode, the system operates as a classical QS, in which a group of channels is taken as a service device.

Mode II of operation, when all possible groups of channels have already been formed, and the system begins to gradually disband them as new requests arrive.

Mode III implies the placing of newly received requests in the queue. The system switches to the classic QS operation mode. Accordingly, the operating mode of the system is determined by the number of requests in it.

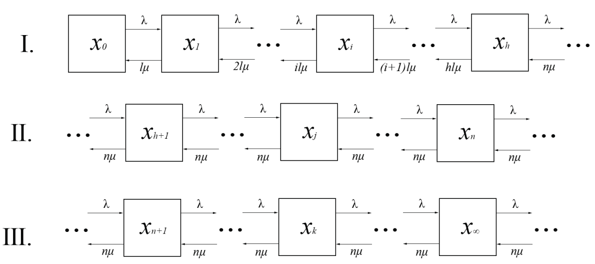

Fig. 2. The state graph of the service model with equal mutual assistance

Let us denote V_{fe} - the number of devices in the system (QS2), l - the number of devices in one group, h - the maximum possible number of groups, μ - the intensity of service of requests on QS2. The ratio of the intensity of incoming requests (λ_{fe}) to the intensity of processing by one group is defined in (8) and the ratio of the intensity of incoming requests to the intensity of processing by all devices of the Front-End system is defined in (9).

$$\alpha = (\frac{\lambda_{fe}}{l \times \mu}) \tag{8}$$

$$\beta = (\frac{\lambda_{fe}}{V_{fe} \times \mu}) \tag{9}$$

The main indicator of the performance of the DPI system is the average time spent by a request in the system (10).

$$\bar{T_{fe}} \approx \frac{(\frac{1}{\sum_{i=0}^{h} i \times l \times \mu + \sum_{j=h+1}^{V_{fe}} j \times \mu} + \frac{\beta}{V_{fe} \times \mu} \times \frac{\alpha^h}{h!} \times \beta \times \frac{1}{(1-\beta)^2})}{(\sum_{i=0}^{h} \frac{\alpha^i}{i!} + \frac{\alpha^h}{h!} \times \frac{\beta^{h+1}}{1-\beta})} \tag{10}$$

So, a dual mathematical model was shown for the DPI system based on various mathematical tools for the hardware filter and Front-End. The initial data for calculating the DPI system have been determined. The formulas for the final processing time of requests (7) and (10) for DPI servers are presented. The sum of the average time for processing an request for QS1 (T_{hw}) (7) and average processing time for a request for QS2 (T_{fe}) (10) gives the average processing time for a request in the DPI system (T_{sys}) - i.e. delay of DPI system corrective actions.

D. Data sets and calculation

In this section, we will briefly present the calculated values of the variables (primarily delay of corrective actions) of the DPI system obtained for a given set of initial data. All values are summarized in the Table II.

To obtain the initial data for the calculations, it was necessary to study the statistics of network traffic that is supposed to be passed through the DPI system. Traffic can be collected and analyzed by wireshark, tshark, cisco NetFlow and others. Traffic recorded by NetFlow will require export to csv or txt formats using Flow-tools. Traffic can be collected and analyzed by wireshark, tshark, cisco NetFlow and others. Traffic recorded by NetFlow will require export to csv or txt formats using Flow-tools. For the calculations presented, we used the traffic collected in the dormitories of Bonch-Bruevich Saint-Petersburg State University of Telecommunications (SPbSUT) using Cisco NetFlow equipment.

TABLE II. INITIAL AND CALCULATED DATA

ACTIONS	NAME	VALUE
Determining the average duration of one flow, s	T_f	300
Determining the number of packets for the period of traffic collection, packets	N_p	245925000
Determining the intensity of arrival packets, packets per second	λ_0	409875
Determining the probability that a flow is known	P_{kn}	0.78
Determination of the average number of packets in a flow, packets	n_{af}	1093
Determining of the average number of remaining requests in QS1	x	27325
Specifying the Hurst Parameter ($0 < H < 1$)	H	0.8
Specifying the characteristic moment of the FBM distribution	a	1
Specifying the number of packets required for analysis on QS2	n_f	10
The choice the coefficient of the operating mode of DPI (= -1 or 0 or 1)	S	1
Specifying bandwidth of DPI system (QS1)	C	829000
Specifying the preliminary number of service devices on QS1	V_{hw}	1
Specifying the preliminary number of service devices on QS2	V_{fe}	1
Specifying the intensity of processing requests (QS2)	μ_{fe}	917
Specifying the permissible maximum delay time for corrective actions (taking into account mode S), s	T_{max}	60
Calculation of the total number of packets of known flows	λ_{old}	319702
Calculation of the total number of packets of new unknown flows	λ_{new}	90173
Calculation of the intensity of incoming requests for QS1 (1).	λ_{hw}	820658
Calculation of the coefficient based on Hurst parameter (6)	$\varphi(H)$	0.6063
Calculation of the value of the QS1 stability condition (4)	ρ_{hw}	0.99
Calculation of the value of the QS2 stability condition (3)	ρ_{fe}	0.99
Calculation of the average processing time of a request on QS1 (7), s	T_{hw}	0.033306
Calculation of the intensity of incoming requests for QS2 from QS1 (2)	λ_{fe}	908
Calculation of the average processing time of a request on QS2 (10), s	T_{fe}	0.10847
Calculation of the delay of corrective actions DPI (sum (7) and (10)), s	T_{sys}	0.142776

The study of traffic on several days during the evening hours (18.00, 21.00, 00.00) showed that the probability of a new flow ($1 - P_{kn}$) appearing varies over time in the range from 6% to 22%. The total number of flows per second ranges from 210 to 240 thousand. For calculations, the average number of flows was taken - 225 thousand. To determine the number of requests remaining in the hardware filter (QS1) for the previous period, a variance of 15 thousand flows was taken.

Hardware filter and Front-End performance was taken to comply with system stability factors. For simplicity, the calculations were performed for one Hardware filter and one Front-End server (which reduces the visibility of the so-called Ventcel-Ovcharov model). The average corrective action delay ($Tsys$) will be the sum of queue delay and processing time on DPI servers, as shown in Table II.

The result of the calculation showed that equipment with a given performance successfully copes with processing the load with a stability coefficient of 0.99. In addition, the hardware filter makes the main contribution to the delay in corrective actions under the current calculated configuration. But an increase in the size of the queue in the hardware filter due to increased load fluctuations can lead to a manifold increase in the delay time for corrective actions. With this delay for corrective actions, there is no need to change server performance. However, it may decrease as the performance of the hardware filter increases.

TABLE III. TIME CHARACTERISTICS OF HARDWARE FILTER AND FRONT-END

MODEL TYPE	HWF SERVICE DEVICE, ITEMS	FE SERVICE DEVICE, ITEMS	TOTAL TIME, ms
SM	1	1	108.9
MM	1	1	142.8
	HWF REQUESTS, ITEMS	**HWF AV. QUEUE, ITEMS**	**HWF AV. TIME REQ. IN SYSTEM, ms**
SM	884128	18905	24.278
MM	847983	-	33.306
	FE REQUESTS, ITEMS	**FE AV. QUEUE, ITEMS**	**FE AV. TIME REQ. IN SYSTEM, ms**
SM	908	63.6	84.593
MM	908	-	108.47

IV. SIMULATION

There are various methods of simulation of environments: GPSS World, network simulator-2 (ns-2), ns-3, OpNet simulator, AnyLogic, Omnet ++, etc.). For the DPI system, a simulation model was created in GPSS [19]. GPSS use a discrete-event approach and a set of distribution laws to describe incoming traffic and how it is processed. In the simulation model, the initial parameters are set, presented in Table III. However, to describe the traffic arrival, the Weibull distribution (for the hardware filter) is used, and for the processing law, the exponential distribution (for the hardware filter and Front-End). It is possible to apply a simulation model in GPSS to obtain the probabilistic-temporal characteristics of the DPI system and compare with the results of the calculation using the mathematical model.

The simulation model of the hardware filter describes the arrival of a request, its spot in a queue, the marking of new requests in the system, the distribution with a given probability for already known flows and for flows requiring analysis. Receiving instructions from Front-End, processing new requests and sending requests to the network, how the end of processing is indicated with different requirements for processing capabilities by a hardware filter and are set when describing the filter.

As a result of the simulation model, you can get the number of requests and data on queues. The number of requests received and processed by the system, requests to the hardware filter and to the Front-End, and responses. The total time spent by all requests in the system (waiting time and processing time), the total number of requests in the queue (without waiting, with waiting). The simulation results are shown in Table III.

The simulation modeling results showed that DPI system hardware can handle the load with a stability coefficient of 0.99. Similar to the results of the mathematical model, in simulation modeling, the hardware filter makes the main contribution to the delay in corrective actions. Changes in the characteristics of the distribution of incoming traffic to the DPI system, which was described in the simulation model by the Weibull distribution, significantly affects the size of the queue, and through it, the processing time of requests in the hardware filter. Comparison of the calculation results based on the mathematical and simulation models given in Table III indicates the possibility of their use. To obtain more accurate results, it is necessary to clarify the parameters of the distribution of the arrival processes and processing of requests in the simulation model.

V. CONCLUSION

The work reported in this paper is a part of the research for the deep packet inspection systems modeling. This article provides a review of the dual mathematical model for calculating the specialized servers of the deep packet inspection system and the delay in corrective actions under the current calculated configuration. The aforementioned work describes formalization of initial data for calculating the deep packet inspection system based on traffic statistical data. That dual mathematical model is based on the modified mathematical model of Norros and also model of Ventcel-Ovcharov with an infinite queue and with equal mutual assistance.

A mathematical model based on FBM was introduced in [2]. In [1], the formulas for the probability of waiting in the queue are given for Norros model. Formulas based on the Norros model for calculating time that request spent in the system were derived in [18]. Mutual assistance models were described in Ventcel and Ovcharov writings [11]–[13]. The Ventcel-Ovcharov model with an infinite queue and with equal mutual assistance is described in [10].

In [19] describes a simulation model was created in GPSS for the DPI system. This simulation model used in order to compare with results of the dual mathematical model.

The results of simulation indicate the possibility of using the dual mathematical model. The possibility of practical application of mathematical and simulation models for calculating the deep packet inspection system is shown.

REFERENCES

[1] C. Grimm and G. Schluchtermann, *IP Traffic Theory and Performance*. Springer, 2008.

[2] I. Norros, "A storage model with self-similar input," *Queueing Syst. Theory Appl.*, no. 16, pp. 387–396, 1994.
[3] ——, "On the use of fractional brownian motion in the theory of connectionless networks," *IEEE J. Sel. Areas Commun.*, no. 13, pp. 953–962, 1995.
[4] M. Krishna, V. Gadre, and U. Desai, *Multifractal Based Network Traffic Modeling*. Springer US, 2003.
[5] T. Aliev, *Basics of modeling discrete systems*. Spb: SPbGU ITMO, 2009.
[6] V. Zaitcev, "Characteristics of the total flow of ip packets at the input of the switching node of a multiservice network," *in Proc. XII conference Technologies of the Information Society*, pp. 178–182, 2018.
[7] L. Kleinrock, "Time-shared system: a theoretical treatment," *J. Assoc. Comput.*, vol. 14, no. 2, pp. 242–251, Mach 1967.
[8] S. Yashkov, "A derivation of responce time distribution for an m/g/1 processor-sharing queue," *Problem. Control Inf. Theory.*, vol. 12, no. 2, pp. 133–148, 1983.
[9] ——, *Analysis of queues in computers*. Radio i Svyaz, Moscow, 1989.
[10] A. Novikov and V. Fitsov, "Application of the ventzel-ovcharov mathematical model with uniform mutual assistance for modern nfv systems," *in Proc. Actual problems of information and telecommunications in science and education VIII Conf.*, vol. 1, pp. 705–709, 2019.
[11] E. Ventzel, *Operations Research*. Soviet Radio, Moscow, 1972.
[12] L. Ovcharov, *Applied problems of queuing theory*. Mashinostroenie, Moscow, 1969.
[13] E. Ventzel, *Probability Theory*. Nauka, Moscow, 1969.
[14] *Recommendation ITU-T Y.2770 Requirements for deep packet inspection in next generation networks*, 2012.
[15] A. Levakov, "Tasks to ensure the functioning of the ngn network in emergency situations," *Vestnik Sviazy*, no. 12, pp. 36–38, 2011.
[16] A. Dainotti, A. Pescape, and C. Sansone, "Early classification of network traffic through multi-classification," *in Proc. Traffic Monitoring and Analysis III*, pp. 122–135, 2011.
[17] J. Park, S. Yoon, and M. Kim, "Software architecture for a lightweight payload signature-based traffic classification system," *in Proc. Traffic Monitoring and Analysis III*, pp. 136–149, 2011.
[18] V. Fitsov, "A mathematical dpi model based on the norros classification," *in Proc. Student Spring - 2018*, pp. 194–200, 2018.
[19] ——, "A simulation model of a dpi system based on gpss world software," *in Proc. Actual problems of information and telecommunications in science and education V Conf.*, pp. 539–545, 2016.