OpenFlow-based Software-Defined Networking Queue Model

Vyacheslav Kartashevskiy and Marina Buranova(✉)

Povolzhskiy State University of Telecommunications and Informatics,
443010 23 L.Tolstoy, Samara, Russia
`buranova-ma@psuti.ru`

Abstract. SDN imposes requirements on manufacturers of infocommunication equipment. It concerns new scenarios support relating to network applications, for example, cloud technologies, the possibility of transmitting large volumes of traffic over long distances. It leads to the need of analyzing features of SDN networks operation, assessing their performance both at design stage and operation process. Quality of service (QoS) parameters in networks can be used as characteristics to assess the quality of operation. In this work, analytical expressions are obtained to estimate the average values of packet delay time in M/G/1 and G/G/1 systems for SDN. For M/G/1 system, solutions are given founded on two approaches. To approximate arbitrary densities (G) in G/G/1 system, an approach based on the use of hyperexponential distributions was used. When analyzing G/G/1 system, it is assumed that there are mutual independence of flows entering the system and the absence of correlations within the sequences of time intervals between packets and packet processing times. The paper presents the result of comparing the estimates of the average values of packet delay time in M/G/1 and G/G/1 systems, where the truncated normal distribution is used as probability density for service time intervals.

Keywords: Queue model · SDN · Quality of service · Hyperexponential distribution · Average package service time in the system

1 Introduction

The development trends of modern infocommunication networks such as the intensive growth of new network applications, the growth of number and types of new network devices, lead to a significant increase in the volume of transmitted traffic. In this case, there is a need to manage heterogeneous flows and provide the required level of QoS while ensuring the security of processing data transmission of flows. This leads to the fact that providers of large networks need to look for new mechanisms for network management, with the ability to quickly configure networks.

Software-Defined Networking (SDN) and Network Function Virtualization (NFV) technologies allow you to control network management using customizable software, making management more intelligent and centralized. This makes the network more flexible, programmable and innovative [1,2].

The introduction of SDN networks entails the need to develop adequate models that allow you to quickly obtain accurate estimates of QoS parameters, which are necessary at the network design stage and further during operation to be able to rapidly respond to changes in network requirements and modification of the network topology [3].

It should be noted that issues of performance and scalability of SDNs are still little studied in terms of constructing analytical models. Simulations and experiments on real networks are of great importance and are widely used to evaluate performance, whereas analytical modeling has its advantages.

Most of the works devoted to the study of SDN operation efficiency, are aimed at developing simulation models and setting up experiments on real equipment. In works [4]-[6] mathematical models are presented for evaluating the performance of controllers and switches under overload conditions. The OpenFlow-based SDN analytical model in [7,8] approximates the data plane as an open Jackson network with a controller modeled as an M/M/1 queue. In [9], a model of SDN performance based on the OpenFlow protocol with several OpenFlow switches is presented, but this model was obtained under the assumption that processed flows are the simplest, while the performance of processing incoming messages of the SDN controller is calculated based on the M/G/1 model. However, it is known [10] that the flows generated by modern applications in the network are not Poisson. That requires taking into account the real parameters of traffic in the applied model.

The works [12,13] present mathematical models for systems that process non-Poissonian flows. It should be noted that only functioning parameters of individual SDN sections are considered, not the network as a whole. The development of SDN analytical model as a $G/G/1$ system remains highly relevant. The technique presented in [9] is convenient to use to expand the packet processing model in the SDN when servicing non-Poisson flows based on the $G/G/1$ mathematical model.

The SDN analytical model based on the OpenFlow protocol is considered below, since it is the most common. Such a model can be used as a basic one for other communication protocols in the SDN. As processed streams, we consider packet traffic generated in the form of bursts, which most closely corresponds to the nature of modern streams formation [14].

Similar to the approach in [9], the process of packet arrival and the procedure for forwarding packets to the switch and the OpenFlow controller are analyzed separately. Then we studied the system for forwarding packets to SDN as a whole. The M/M/1 system is considered first, followed by the G/G/1 system.

2   Model of Queuing of OpenFlow Switches

In general, the SDN network can be represented in the form of a diagram shown in Fig. 1.

All analytical models of OpenFlow networks, developed so far [9, 12, 13], are based on the assumption that packet traffic to the SDN switch corresponds to the Poisson distribution. At the same time, traffic studies in [10, 15, 16] showed that the arrival of packets in infocommunication networks has a burst character and differs from the Poisson flow [17].
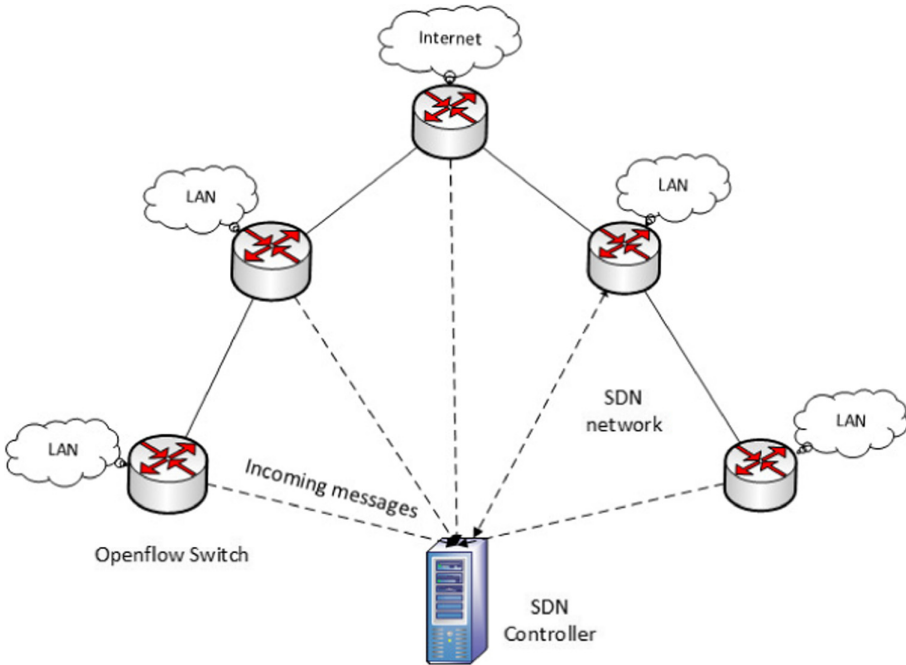

Fig. 1. Typical OpenFlow Network Scenario.

The procedure of forwarding packets to the switch is illustrated in Fig. 2. Each received packet is analyzed by the switch in order to identify a flow in regard to find the assigned rule for its processing or forwarding in its tables. If no forwarding entry is found for this stream, the switch sends a message containing the complete packet or its identifier to the SDN controller. The controller analyzes the packet, sets the flow processing rule to which the packet belongs, and sends the set rule to the switch to be added to its flow tables.

Since requests to the forwarding tables for all packets are independent of each other, the packet processing time can be represented as a random variable with an exponential distribution. Then, provided there is sufficient packet buffer

capacity for all incoming queues in the switch, the packet forwarding queue model in the OpenFlow switch can be represented as the model of the queue type $M/M/1$ [9]. Further, this approach can be developed to the $G/G/1$ system.

To describe the queue in the OpenFlow switch, we introduce the following notation: $\lambda_i^{(b)}$ is the intensity of the arrival of the Poisson stream of packets in a batch to the i-th OpenFlow switch, $\lambda_i^{(p)}$ is intensity, characterizing the Poisson distribution of the number of packets in a batch,) $\mu_i^{(s)}$ is the intensity of processing the s-th packet by the switch (corresponds to the exponential distribution).

Suppose that a burst of m packets arrives at the i-th ($1 \leq i \leq k$) OpenFlow switch, where n packets are waiting for their turn for processing. The incoming l-th packet must wait until n waiting packets in the buffer and the first (l–1) packets from the same batch are processed. Based on these assumptions for the $M/M/1$ system, we can obtain the ratio between the average queue length and the average time spent by a packet in the queue [9].

$$\overline{L_i^{(s)}} = \sum_{n=0}^{\infty} np_n = \lambda_i^{(b)}\lambda_i^{(p)}\overline{W_i^{(s)}}. \tag{1}$$

The average time of a packet's stay in the switch is determined in a similar way.

$$\overline{W_i^{(s)}} = \frac{\lambda_i^{(p)} + 1}{2\left(\mu_i^{(s)} - \lambda_i^{(b)}\lambda_i^{(p)}\right)} \ . \tag{2}$$

And based on (1) and (2), you can write an expression to estimate the average length of the switch queue

$$\overline{L_i^{(s)}} = \frac{\lambda_i^{(b)}\lambda_i^{(p)}\left(\lambda_i^{(p)} + 1\right)}{2\left(\mu_i^{(s)} - \lambda_i^{(b)}\lambda_i^{(p)}\right)} \ . \tag{3}$$

3   SDN Controller Queuing Model

Now let's look at the procedure of forwarding packets to the OpenFlow controller. Upon arrival of a new flow, the OpenFlow switch sends an incoming packet message as a request to configure flow processing to its SDN controller (request packet). In this regard, the incoming flow of messages from the switch to its controller corresponds to the process of the flow entering the switch. In an OpenFlow network, the controller is usually responsible for several switches and receives a stream of incoming request packets from each of them.

Based on the assumption that the arrival process on the i-th switch is Poisson with the parameter $\lambda_i^{(f)}$, and the processes on all k switches are independent of each other, we can further conclude that the total flow of message packets to the controller is also Poisson ...
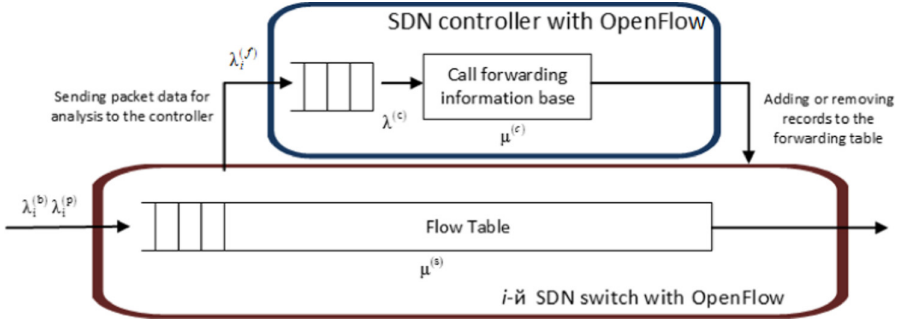
Fig. 2. SDN Controller Inbound Processing.

For an SDN controller responsible for k OpenFlow switches, the flow arriving at the i-th ($i = 0, 1, 2, ..., k$) switch is a Poisson flow with the parameter $\lambda_i^{(f)}$, independent of messages on other switches, then all packets of incoming messages from k switches to the controller correspond to the Poisson distribution with the parameter $\lambda_{(c)}$.

$$\lambda_{(c)} = \sum_{i=1}^{10} \lambda_i^{(f)}. \tag{4}$$

Upon arrival of a request packet from the switch, the controller determines the flow transfer rule to which the packet that has come from the switch belongs. To do this, the controller processing unit reads the incoming packet message from the queue after the last message has been processed. The incoming packet message is then processed by looking up the forwarding information base (FIB). The FIB entry contains routing information obtained using routing protocols or manual programming. Finally, the controller encapsulates the found forwarding rule in a packet message for the appropriate switch.

When processing an incoming packet-request from the switch by the SDN controller, the processing time of the incoming packet message is mainly determined by FIB lookups, which are performed by matching the longest prefix. In this regard, in [9,18], it was assumed that the search time in the FIB is considered to be distributed normally with the mean value $1/\mu^{(c)}$ and variance $\sigma^{(c)}$. The parameter $\mu^{(c)}$ is the average processing rate of incoming packet messages in the controller, and the parameter $\sigma^{(c)}$ is the standard deviation. The queue in the controller is organized according to the FIFO (First In - First Out) type, and the arrival and processing of incoming message packets are independent of each other.

In accordance with the assumptions made, it is possible to characterize the processing of message packets by the SDN controller using the $M/N/1$ queuing model, where the symbol N corresponds to a normal distribution.

To analyze a queue in a controller, it is convenient to use a semi-Markov process as a model, in which state changes occur at the moment the packet

leaves. For such moments, the nested Markov chain should be defined as the number of claims present in the system at the moment the next serviced claim leaves. In this case, you can use the approach shown in [19], which is based on the application of the method of additional variables. It should be noted that the same technique was used to analyze the queue in the SDN controller in [9].

The length of the packet message queue in the controller at time t is denoted by $L(t)$. For any fixed time t, if there is a processed packet message, the distribution of the remaining processing time does not depend on time t, and the queue length $L(t), t \leq 0$ no longer has the properties of a Markov stream. Suppose that v_n is the number of incoming message packets in the controller during the processing time x_n of the n-th message. Then $\{v_n, n \geq 1\}$ is a built-in Markov chain.

If t_n is the required processing time of the (n+1)-th message, for any t_n the probability density function, in accordance with the above assumptions, can be written as

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma^{(c)}} e^{\frac{\left(x - 1/\mu^{(c)}\right)^2}{2\sigma^{(c)2}}}.$$ (5)

Let us denote by η_n the number of new packets arriving at the controller during processing of the $(n+1)$-th packet under the assumption that the arrival process is Poisson. Then

$$P\left(\eta_n = k\right) = P\left(\eta_n = k | t_n = x\right) f(x)\mathrm{d}x = \int_0^\infty \frac{(\lambda^{(c)}x)^2}{k!} e^{-\lambda^{(c)}x} f(x)\mathrm{d}x,$$ (6)

where $P\left(\eta_n = k | t_n = x\right)$ is the probability of one-step transition, $k = 0, 1, 2, ...$.

We set that $a_k = P(\eta_n = k) > 0$. Then $\{v_n, n \geq 1\}$ constitutes a Markov chain, the state change diagram of which is shown in Fig. 3, and the transition matrix will have the following form [9,19]

$$P = \begin{pmatrix} a_0 & a_1 & a_2 & \cdots \\ a_0 & a_1 & a_2 & \cdots \\ 0 & a_0 & a_1 & \cdots \\ 0 & 0 & a_1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

Using the apparatus of generating functions for processing the sequences v_n and η_n, according to [9,19], for the average queue length of messages from the switch to the controller in the $M/G/1$ system, we can obtain

$$\overline{L^{(c)}} = \rho^{(c)} + \frac{\rho^{(c)2} + \lambda^{(c)2}\sigma^{(c)2}}{2(1 - \rho^{(c)})},$$ (7)

where $\rho^{(c)}$ is the load factor of the controller.

For the average residence time of packet messages in the controller, the expression will look like
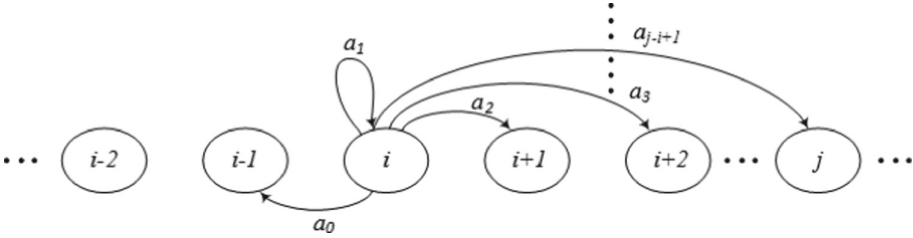
Fig. 3. Transition probability diagram for an embedded Markov chain of type M/G/1.

$$\overline{W^{(c)}} = \frac{\overline{L^{(c)}}}{\lambda^{(c)}} = \frac{1}{\mu^{(c)}} + \frac{\rho^{(c)^2} + \lambda^{(c)^2}\sigma^{(c)^2}}{2\lambda^{(c)}(1 - \rho^{(c)})}. \tag{8}$$

4   SDN Network Queuing Model with OpenFlow

On OpenFlow networks, the switch maintains a queue for all packets arriving on any ingress port and forwards them according to its internal flow tables. If there is no entry in the tables corresponding to the packet, that is, the packet belongs to a new flow, the switch will send a request for flow processing rules to its SDN controller and, having received the corresponding rule from the controller, sends the packets of this flow in accordance with this rule. At the same time, the controller keeps a queue for all request packets from its slave switches. Using the above analysis of the service process for OpenFlow switches and SDN controllers, we can represent the packet processing model in OpenFlow networks as a queuing system according to Fig. 2.

In Fig. 2, the i-th OpenFlow switch with rate $\mu^{(s)}$ processes bursts of packets arriving at rate $\lambda^{(p)}\lambda^{(p)}$. Suppose that the incoming packet in the i-th switch belongs to a new flow with probability q_i, the switch sends a request packet at a rate $\lambda_i^{(f)}$ to its controller SDN. The controller receives request packets with a rate $\lambda^{(c)}$ from k OpenFlow switches with a $\mu^{(c)}$ and processes them. Replies to request packets are sent back to forward and update the flow tables in the corresponding switch at a rate

$$\lambda_i^{(f)} = q_i\lambda_i^{(b)}\lambda_i^{(p)}. \tag{9}$$

Let W_i be the time of packet forwarding through the i-th OpenFlow switch, you can calculate it taking into account two possible processing cases: direct forwarding and forwarding with the participation of the controller. The packet forwarding time in the latter case consists of two parts: the packet sojourn time in the switch $W_i^{(s)}$ and the sojourn time of the corresponding packet transmission message in the $W^{(c)}$ controller.

$$W_i = \begin{cases} W_i^{(s)} & \text{with probability } 1 - q_i \\ W_i^{(s)} + W^{(c)} & \text{with probability } q_i. \end{cases} \tag{10}$$

The average time to process a packet in SDN can be determined by the average time to forward packets through the OpenFlow switch $(\overline{W_i^{(s)}})$ according to the diagram shown in Fig. 2.

5   Results for the System $M/G/1$

It can be shown [9] that the average time for forwarding packets through the OpenFlow switch $\overline{W_i}$, where the waiting times $W_i^{(s)}$ and $W^{(c)}$ from (10) can be obtained from expressions (6) and (8) respectively, finally they will have the form

$$\overline{W_i} = E\left[W_i\right] = (1 - q_i)\, E\left[W_i^{(s)}\right] + q_i\left(E\left[W_i^{(s)}\right] + E\left[W^{(c)}\right]\right)$$
$$= \overline{W_i^{(s)}} + q_i\overline{W_i^{(c)}} \tag{11}$$
$$= \frac{\lambda_i^{(p)}+1}{2(\mu_i^{(s)})-\lambda_i^{(p)}\lambda_i^{(c)}} + q_i\Big(\frac{1}{\mu_i^{(c)}} + \frac{\rho^{(c)2}+\lambda^{(c)2}\sigma^{(c)2}}{2\lambda^{(c)}(1-\rho^{(c)})}\Big),$$

where $\delta_i^{(s)}$ and $\delta^{(c)}$ are the service rates in the i-th switch and controller, respectively.

The result for estimating the average packet processing time in SDN, obtained in the form of formula (11), corresponds to the mathematical model $M/N/1$.

Let us introduce the notation $\delta_i^{(s)} = 1/\overline{W_i^{(s)}}$ for the i-th switch and $\delta^{(c)} = 1/\overline{W^{(c)}}$ for the controller, where $\delta_i^{(s)}$ and $\delta^{(c)}$ are the parameters, which are defined as [19], ξ is the root of the equation $\xi = \Lambda_V(\mu - \mu\xi)$.

Here Λ_V is the Laplace transform of the density $f_V(\cdot)$, μ is the average intensity of packet processing in the system, and $\delta_i{}^{(s)}$ and $\delta^{(c)}$ are the parameters of the density of the distribution of processing time in the queue in the i-th switch and controller, respectively. The method for determining these parameters is well described in [19–21].

Based on the above, more general results for the M/G/1 system can be obtained. The expressions for the probability density of the packet processing time S in the controller $f_{(S)i}(\cdot)$ and the packet processing time C in the switch $f_{(C)}(\cdot)$ are represented in the form

$$f_{(S)i}(u) = \delta_i^{(s)} e^{-\delta_i^{(s)}u}, \tag{12}$$

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma^{(c)}} e^{-\frac{\left(u-1/\delta^{(c)}\right)^2}{2\sigma^{(c)2}}}. \tag{13}$$

Assuming that C and S are independent, and taking into account (12) and (13) in expression (10), we obtain

$$w_{(S,C)i}(u) = (1 - q_i)\, f_{(S)i}(u) + q_i\left[f_{(S)i}(u) \odot f_{(C)}(u)\right], \tag{14}$$

where $\odot$ is the convolution symbol The second term in expression (14) can be represented as

$$q_i \int_0^u \frac{1}{\sqrt{2\pi}\sigma^{(c)}} e^{-\frac{\left(x-1/\delta^{(c)}\right)^2}{2\sigma^{(c)2}}} \cdot \delta_i^{(s)} \cdot e^{-\delta_i^{(s)}(u-x)} \mathrm{d}x. \tag{15}$$

The upper limit of integration in (15) is u that is equal to infinity for the normal distribution and is equal to some finite value (the maximum value of the packet duration in the controller) for the truncated normal distribution.

For the $M/G/1$ system, where the normal distribution is used as an example of an arbitrary distribution, that is, for the $M/N/1$ system, the expression for the probability density of the packet processing time in the system can be written as:

$$w_{(S,C)i}(u) = (1 - q_i)\, \delta_i^{(s)} e^{-\delta_i^{(s)} u}$$
$$+q_i \frac{\delta_i^{(s)}}{2R} e^{-\delta_i^{(s)} u} e^{((\frac{1}{\delta^{(c)}})^2)/2\sigma^{(c)2}) + \sigma^{(c)2}\delta_i^{(s)2} - 2\delta_i^{(s)}\frac{1}{\delta^{(c)}}} \tag{16}$$
$$\times [1 - \Phi(\frac{\sigma^{(c)}}{\sqrt{2}}(\delta_i^{(s)} - \frac{1}{\delta^{(c)}\sigma^{(c)2}}))],$$

where $\Phi(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} \mathrm{d}t$ [22],

$R = \frac{1}{2}[1 + \Phi(\frac{1}{\sqrt{2}\delta^{(c)}\sigma^{(c)}})]$ is normalizing constant, which is determined from the condition $\int_0^T f(u)\mathrm{d}u = 1$. For a truncated normal distribution

$R = \frac{1}{2}\Phi(\frac{\frac{1}{\delta^{(c)}}\sigma^{(c)}}{\sqrt{2}})$

Estimating the values of the integral function is associated with some difficulties that lead to the need to use various methods of approximation in the form of infinite power series [21], infinite continued fractions [23], polynomials of a special form [23] and empirical formulas [24]. In the case of expression (16), the values of the function $\Phi(x)$ can be determined in a table.

Let us compare the time densities of packet processing in the $H_2/H_2/1$ system according to (16) for different distribution parameters. The comparison result is shown in Fig. 4.

From a comparison of the graphs shown in Fig. 4, it follows that for the values of the distribution parameters $\delta_i^{(s)} = 1$ and $\delta^{(c)} = 1$, the distribution degenerates into an exponential, in other cases an insignificant peak appears.

The final expression for estimating the average packet processing time in the $M/N/1$ system for SDN will have the form

$$\bar{W}_{(S,C)i}(u) = (1 - q_i)\frac{1}{\delta_i^{(s)}} + q_i M \frac{1}{\delta_i^{(s)2}}, \tag{17}$$

where $M = \frac{\delta_i^{(s)}}{2R} \cdot e^{((\frac{1}{\delta^{(c)}})^2/2\sigma^{(c)2}) + \sigma^{(c)2}\delta_i^{(s)2} - 2\delta_i^{(s)}\frac{1}{\delta^{(c)}}}[1 - \Phi(\frac{\sigma^{(c)}}{\sqrt{2}}(\delta_i^{(s)} - \frac{1}{\delta^{(c)}\sigma^{(c)2}}))]$.

Taking into account formulas (1) and (17), the expression for estimating the average queue length will be written in the form

$$L = \lambda_i^{(b)}\lambda_i^{(p)} \cdot (1 - q_i)\frac{1}{\sigma_i^{(s)}} + q_i M \frac{1}{\sigma_i^{(s)2}}. \tag{18}$$
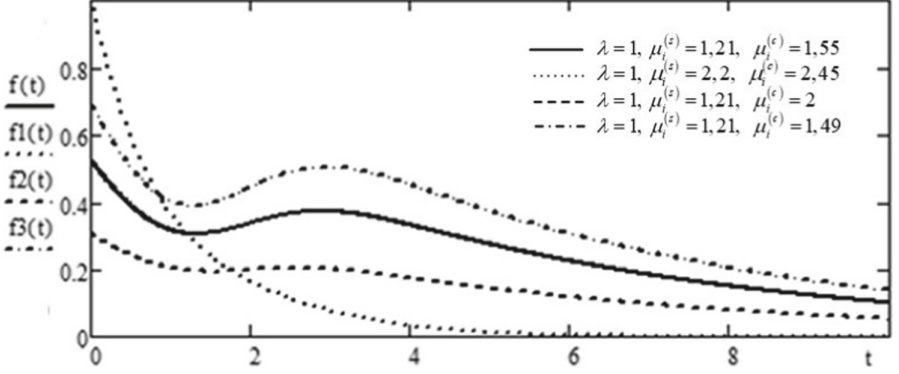
Fig. 4. Probability densities of packet processing times in SDN according to the formula (16): $f(t) - \delta_i^{(s)} = 0,2, \sigma^{(c)} = 0,5$; $f1(t) - \delta_i^{(s)} = 1, \sigma^{(c)} = 1$; $f2(t) - \delta_i^{(s)} = 0,2, \sigma^{(c)} = 0,8$; $f3(t) - \delta_i^{(s)} = 0,2, \sigma^{(c)} = 0,45$.

This approach allows us to evaluate the performance of the SDN network and the main parameters of the quality of service of traffic in the SDN network in the case of processing Poisson streams. Given that modern applications generate non-Poissonian traffic, queuing systems are better described by $G/G/1$ models.

6   SDN Network Queuing Model with OpenFlow

It is interesting to compare the values of the estimates of the SDN functioning parameters obtained for the $M/G/1$ and $G/G/1$ systems. Obviously, for this, one should use the result shown in (16), (17), and (18) for the $M/N/1$ system and find an expression for the packet waiting time density in the $G/G/1$ system. As an example of an arbitrary probability distribution for the query processing time in the controller, similarly to the approach shown for the $M/N/1$ system, we can use the normal distribution (for the random variable C) (15), and for the random variable S, we can use the probability density as an approximation by hyperexponential distribution:

$$f_{(S)i}(u) = p\delta_{1i}^{(s)} e^{-\delta_{1i}^{(s)} u} + (1-p)\delta_{2i}^{(s)} e^{-\delta_{2i}^{(s)} u}, \tag{19}$$

where p, $\delta_{1i}^{(s)}$, $\delta_{2i}^{(s)}$ is distribution parameters. Then the $G/G/1$ model will be approximated by the $H_2/N/1$ model, and expression (16) takes the form

$$
\begin{aligned}
w_{(S,C)i}(u) = (1-q_i)\left(p\delta_{1i}^{(s)} e^{-\delta_{1i}^{(s)} u} + (1-p)\delta_{2i}^{(s)} e^{-\delta_{2i}^{(s)} u}\right) \\
+ q_i\left\{M_1 e^{-\delta_{1i}^{(s)} u} + (1-p) M_2 e^{-\delta_{2i}^{(s)} u}\right\},
\end{aligned}
\tag{20}
$$

where $M_1 = \frac{\delta_{1i}^{(s)}}{2R} \cdot e^{((\frac{1}{\delta^{(c)}})^2/2\sigma^{(c)2}) + \sigma^{(c)2}\delta_{1i}^{(s)2} - 2\delta_{1i}^{(s)}\frac{1}{\delta^{(c)}}}[1 - \Phi(\frac{\sigma^{(c)}}{\sqrt{2}}(\delta_{1i}^{(s)} - \frac{1}{\delta^{(c)}\sigma^{(c)2}}))]$,

$M_2 = \frac{\delta_{2i}^{(s)}}{2R} \cdot e^{((\frac{1}{\delta^{(c)}})^2/2\sigma^{(c)2}) + \sigma^{(c)2}\delta_{2i}^{(s)2} - 2\delta_{2i}^{(s)}\frac{1}{\delta^{(c)}}}[1 - \Phi(\frac{\sigma^{(c)}}{\sqrt{2}}(\delta_{2i}^{(s)} - \frac{1}{\delta^{(c)}\sigma^{(c)2}}))]$.

The value of the average packet processing time in the system when using the approximation by the hyperexponential distribution of the distribution density of the packet processing time in the switch can be written as:

$$\bar{W}_{(S,C)i}(u) = (1-q_i)\left(\frac{p}{\delta_{1i}^{(s)}} + \frac{(1-p)}{\delta_{2i}^{(s)}}\right) + q_i\left[M_1\frac{p}{\delta_{1i}^{(s)2}} + M_2\frac{(1-p)}{\delta_{2i}^{(s)2}}\right]. \quad (21)$$

Similar to that obtained in (18) and taking into account (19), the expression for estimating the average queue length is written as:

$$\bar{L} = \lambda_i^{(b)}\lambda_i^{(p)}\left\{(1-q_i)\left(\frac{p}{\delta_{1i}^{(s)}} + \frac{(1-p)}{\delta_{2i}^{(s)}}\right) + q_i\left[M_1\frac{p}{\delta_{1i}^{(s)2}} + M_2\frac{(1-p)}{\delta_{2i}^{(s)2}}\right]\right\}. \quad (22)$$

To obtain a more general result, the approximation of arbitrary densities in the $G/G/1$ system is preferable to be represented in the form of hyperexponential distributions for random variables S and C, that is, it is necessary to use expression (19) for S, and for C

$$f_{(C)}(u) = g\delta_1^{(c)}e^{-\delta_1^{(c)}u} + (1-g)\delta_2^{(c)}e^{-\delta_2^{(c)}}, \quad (23)$$

where g, $\delta_1^{(c)}$, $\delta_2^{(c)}$ is packet processing time density parameters in the controller. After such a replacement, the system $G/G/1$ will be approximated by the system $H_2/H_2/1$.

At the same time, for the density of the packet processing time in the $H_2/H_2/1$ system in the SDN network, one can obtain

$$f(x) = \frac{1}{(\sqrt{2\pi})\sigma^{(c)}}e^{\frac{\left(x-1/\mu^{(c)}\right)^2}{2\sigma^{(c)2}}}, \quad (24)$$

where $A = \frac{pg\delta_{1i}^{(s)}}{\delta_{1i}^{(s)}-\delta_1^{(c)}}$, $B = \frac{(1-p)g\delta_{2i}^{(s)}}{\delta_{2i}^{(s)}-\delta_1^{(c)}}$, $L = \frac{p(1-g)\delta_{1i}^{(s)}}{\delta_{1i}^{(s)}-\delta_2^{(c)}}$, $D = \frac{(1-p)(1-g)\delta_{2i}}{\delta_{2i}^{(s)}-\delta_2^{(c)}}$.

Using (24), the analytical expression for the average packet processing time in the $H_2/H_2/1$ system for the SDN network can be written as:

$$\bar{W}_{(S,C)i}(u) = (1-q_i)\left[\frac{p}{\delta_{1i}^{(s)}} + \frac{(1-p)}{\delta_{2i}^{(s)}}\right] + q_i\left[\frac{(A+B)}{\delta_1^{(c)}} + \frac{(L+D)}{\delta_2^{(c)}}\right]. \quad (25)$$

From (25), the value of the average queue length in the SDN is written as

$$\bar{L} = \lambda_i^{(b)}\lambda_i^{(p)}\left\{(1-q_i)\left[\frac{p}{\delta_{1i}^{(s)}} + \frac{(1-p)}{\delta_{2i}^{(s)}}\right] + q_i\left[\frac{(A+B)}{\delta_1^{(c)}} + \frac{(L+D)}{\delta_2^{(c)}}\right]\right\}. \quad (26)$$

The delay variation can be determined taking into account (25) in the form of the expression

$$\sigma_{(S,C)i}(u) = \sqrt{\begin{array}{l}(1-q_i)\left[p\left(\frac{2}{\delta_{1i}^{(s)2}} - \frac{1}{\delta_{1i}^{(s)}}\right) + (1-p)\left(\frac{2}{\delta_{2i}^{(s)2}} - \frac{1}{\delta_{2i}^{(s)}}\right)\right] + \\ +q_i\left[(A+B)\left(\frac{2}{\delta_1^{(c)2}} - \frac{1}{\delta_1^{(c)}}\right) + (L+D)\left(\frac{2}{\delta_2^{(c)2}} - \frac{1}{\delta_2^{(c)}}\right)\right].\end{array}} \quad (27)$$

As a result, formulas were obtained for the density of distributions of packet service times in the SDN network, if the queuing system is represented by the G/G/1 model for two cases:

1) the distribution density of the packet processing time in the switch is approximated by a hyperexponential distribution, the distribution density of the packet processing time in the controller is estimated by a normal distribution (the $M/H_2/1$ system);
2) the distribution density of packet processing times in the switch and controller is approximated by a hyperexponential distribution (the $H_2/H_2/1$ system).

7   Numerical Results

Formula (17) allows us to estimate the average packet processing time in the G/N/1 system for the SDN network; earlier in [9], formula (11) was obtained to approximate the average packet processing time in the M/N/1 system. To determine the accuracy of the estimates obtained, it is necessary to compare the values accepted for various models, including the results received for $M/N/1$ and $H_2/N/1$.

Let us compare the numerical estimates of the average processing times in the M/N/1 system for these two approaches. We will accept the following conditions for the operation of the network: $\lambda_i^{(b)}\lambda_i^{(p)} = 0,5$, $\lambda_i^{(p)} = 1$, $\mu^{(c)} = 1$, $\sigma^{(c)} = 1$ number of switches is 1.

The values of the functioning parameters - the average packet processing time and the average queue length for SDN, are presented in the table (Table 1).

Figure 5 shows the dependence of the delay in the SDN network depending on the network load factor for the cases considered.

Obviously, the average packet processing time in the SDN network is expected to grow with the increase in the network load factor.

Table 1. The values of the functioning parameters.

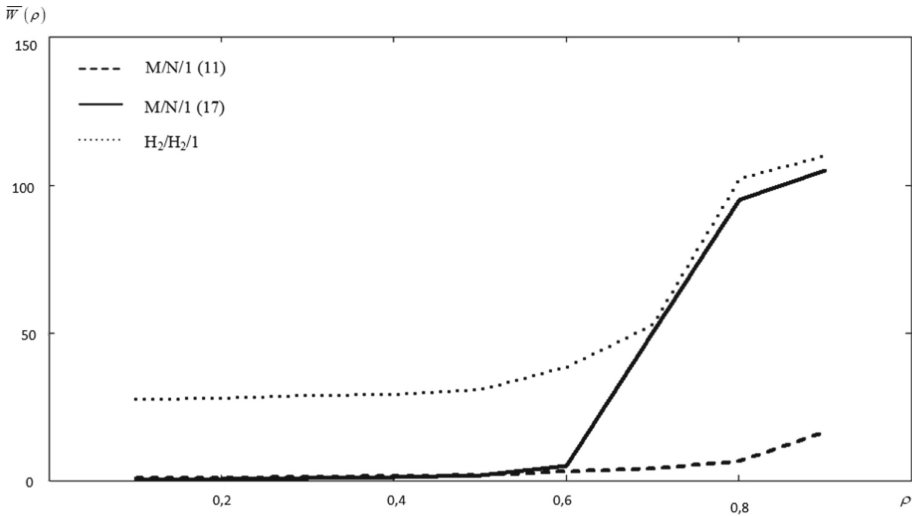| Parameter values	$\bar{W}_{(S,C)i}$, conditional units of time	$\bar{L}$, conditional units of time
System $M/N/1$ (11)	2.3	0,23
System $M/N/1$ (17)	2,9	0,29
System $H_2/N/1$ (21)	1,5	0,15
System $H_2/H_2/1$ (25)	29,1	2.9

$\overline{W}(\rho)$

Fig. 5. The graph of the dependence of the delay on the load factor in the SDN network.

8   Conclusion

In the process of designing, deploying and operating networks, it is necessary to take into account the quality of service parameters of the flows processed in the network. In this paper, a model of the SDN network functioning is built on the basis of the mathematical apparatus of the queuing theory. As a result, analytical expressions were obtained for evaluating the main parameters of traffic QoS in SDN for the M/G/1 system and for the $G/G/1$ system, provided that the flows entering the system are mutually independent and there are no correlations within the sequences of time intervals between packets and processing times packages.

In the development of this work, it is planned to build an SDN model for the G/G/1 system when processing correlated streams. The problem of accounting for correlations within flows will improve the accuracy of the analytical model, which can further help in the development of accurate simulation models of the functioning of SDN networks. This will make it possible to quickly analyze the efficiency of the SDN network in real operating conditions.

References

1. Feamster, N., Rexford, J., Zeguraz, E.: The road to SDN: an intellectual history of programmable networks. Networks **11**(12), 20–40 (2013)
2. Astutoz, B.N., Mendonca, M.: A survey of software-defined networking: past, present, and future of programmable networks. IEEE Commun. Surv. Tutor. **16**(3), 1617–1634 (2014)

3. McKeown, N., et al.: Openow: enabling innovation in campus networks. ACM SIGCOMM Comput. Commun. Rev. (CCR) **38**(2), 69–74 (2008)
4. Bozakov, Z., Rizk, A.: Taming SDN controllers in heterogeneous hardware environments. In: Proceedings of the Second European Workshop on Software Defined Networks (EWSDN), pp. 50–55 (2013)
5. Azodolmolky, S., Wieder, P., Yahyapou, R.: Performance evaluation of a scalable software-defined networking deployment. In: Proceedings of the Second European Workshop on Software Defined Networks (EWSDN), pp. 68–74 (2013)
6. Azodolmolky, S., Nejabati, R., Pazouki, M., Wieder, P.: An analytical model for software defined networking: a network calculus-based approach. In: Proceedings of the 2013 IEEE Global Communications Conference (GLOBECOM), pp. 1397–14021 (2013)
7. Jarschel, M., Oechsner, S., Schlosser, D., Pries, R., Goll, S., Phuoc, T.G.: Modeling and performance evaluation of an openflow architecture. In: Proceedings of the Twenty-Third International Teletraffic Congress (ITC), pp. 1–7 (2011)
8. Mahmood, K., Chilwan, A., Osterbo, O., Jarschet, M.: Modelling of open-flow-based software-defined networks: the multiple node case. IET Netw. **4**(5), 278–284 (2015)
9. Xiong, B., Yang, K., Zhao, J., Li, W., Li, K.: Performance evaluation of OpenFlow-based software-defined networks based on queueing model. Comput. Netw. **102**, 174–183 (2016)
10. Sheluhin, O.I., Osin, A.V., Smolskij, S.M.: Samopodobie i fraktaly. Telekommunikacionnye prilozheniya, Fizmatlit (2008)
11. Taggu, M.S.: Self-similar processes. In: Kotz, S., Johnson, N. (eds.) Encyclopedia of Statistical Sciences, pp. 352–357. Wiley, New York (1988)
12. Samujlov, K.E., Shalimov, I.A., Buzhin, I.G., Mironov, Y.B.: Model funkcionirovaniya telekommunikacionnogo oborudovaniya programmno-konfiguriruemyh setej. Sovremennye informacionnye tekhnologii i IT-obrazovanie **14**(1), 13–26 (2018)
13. Muhizi, S., Shamshin, G., Muthanna, A., Kirichek, R., Vladyko, A., Koucheryavy, A.: Analysis and performance evaluation of SDN queue model. In: Koucheryavy, Y., Mamatas, L., Matta, I., Ometov, A., Papadimitriou, P. (eds.) WWIC 2017. LNCS, vol. 10372, pp. 26–37. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61382-6_3
14. Okamura, H., Dohi, T., Trivedi, K.S.: Markovian arrival process parameter estimation with group data. ACM Trans. Netw. **17**(4), 1326–1339 (2009)
15. Jain, R., Routhier, S.A.: Packet trains: measurements and a new model for computer network traffic. IEEE J. Sel. Areas Commun. **4**(6), 986–995 (2016)
16. Choi, B.D., Choi, D.I., Lee, Y.D., Sung, K.: Priority queueing system with fixed-length packet-train arrivals. In: Fifth International Conference on Information Technology: New Generations (ITNG 2008), vol. 145, no. 5, pp. 331–336 (1998)
17. Paxson, V., Floyd, S.: Wide area traffic: the failure of Poisson modeling. IEEE/ACM Trans. Netw. **3**(3), 226–244 (1995)
18. Liang, Z., Xu, K., Wu, J.: A novel model to analyze the performance of routing lookup algorithms. In: Proceeding of the 2003 IEEE International Conference on Communication Technology (ICCT), pp. 508–513 (2003)
19. Kleinrock, L.: Queueing Systems. Vol. I: Theory. Wiley, New York (1975)
20. Tarasov, V., Kartashevskiy, I.: Approximation of input distributions for queuing system with hyper-exponential arrival time. In: 2015 Second International Scientific-Practical Conference Problems of Infocommunications Science and Technology, pp. 15–17. IEEE, Kharkiv (2015)

21. Kartashevskiy, I., Buranova, M.: Calculation of packet jitter for correlated traffic. In: Galinina, O., Andreev, S., Balandin, S., Koucheryavy, Y. (eds.) NEW2AN/ruSMART -2019. LNCS, vol. 11660, pp. 610–620. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30859-9_53
22. Gradshtejn, I.S., Ryzhik, I.M.: Tablicy integralov, summ, ryadov i proizvedenij. 4-e izd. Fizmatgiz (1963)