INFORMATION TECHNOLOGY, COMPUTER SCIENCE AND MANAGEMENT
ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ

UDC 621.395.4
https://doi.org/10.23947/2687-1653-2026-26-2-2221

Original Theoretical Research

EDN: MPFRRV

Analytical Model of the Buffer Memory of an OpenFlow Switch in a Software-Defined Network (SDN)

Gennady V. Slyusarev, Valery P. Mochalov, Dar'ya V. Gosteva ✉
North-Caucasus Federal University, Stavropol, Russian Federation
✉ DVGus@yandex.ru

Abstract

Introduction. Reliable identification of the probabilistic and temporal features of switching nodes is required for assessing the quantitative characteristics of software-defined networks. Widely used queuing theory (QT) methods only approximately specify and model the processes in an OpenFlow switch and its buffer memory. This results in understated and unrealistic performance estimates for the designed network equipment, causing switch buffer overloading and packet loss. A different modeling approach can solve this problem. The objective of this paper is to develop and study an analytical model for the buffer memory of an OpenFlow switch in an SDN using advanced techniques.

Materials and Methods. The discrete Laplace-Stieltjes transform was used. Statistical characteristics of packet flows and the throughput of communication channels for a given packet loss probability were taken into account. The OpenFlow switch buffer memory model was based on the mathematical apparatus of the QT. It was constructed under the assumption of recurrence of input data flows with batch arrivals. The model was based on schematic representations of the switch structure, its record set, and a graph description of the transmission of network packets leaving the switch. We started with schematic representations of the switch structure, its record set, and a graph description of the transmission of network packets exiting the switch. Two model assumptions were taken as acceptable:

– arbitrary distribution of the relationship between the volume of data flows and their service time;
– discreteness of the distribution of the information flow structure.

Results. The developed model integrated the probability of packet flow loss, their statistical characteristics, the throughput of computing devices, and the multiphase service procedure. When testing the model performance, we assumed that the switch load increased from 0.1 to 0.9, and the loss probability — from 10^{-3} to 10^{-6}. For these metrics, we determined how the switch load affected the buffer memory size and latency. In the first case, the minimum value (memory capacity) was 0.201, the maximum — 10564. In the second, they were 0 and 470 ms, respectively. For simulation modeling, the minimum time was 0 ms, the maximum — 2300 ms. The simulation and analytical modeling indicators were close at loads below 50% and increased several times at loads above 50%. The indicators increased sharply with loads up to 70%, and then increased exponentially.

Discussion. At low network loads, queues did not overflow, packets were not lost, and linear dependences were maintained. At medium and high loads, packet flow processing was described by nonlinear dependences. The results of analytical and simulation modeling diverged due to the explosive nature of self-similar network traffic and its approximate description by the Pareto distribution. Switch load determined the feasibility of the proposed approach. The model is suitable for designing elements of software-defined networks to analyze their resilience under various information impacts.

Conclusion. The proposed SDN analytical model determined the values and variances of the switch buffer memory size, as well as the memory capacity for constructing address flow tables. The solution performance was tested with switch loads ranging from 0.1 to 0.9. It is planned to create a model that takes into account request flows from both the external network and the server.

Keywords: improving queueing theory methods, OpenFlow buffer memory analytical model, network packet loss, switch simulation modeling

Acknowledgements. The authors would like to thank the Editorial board and the reviewers for their attentive attitude to the article and for the specified comments that improved its quality.

For Citation. Slyusarev GV, Mochalov VP, Gosteva DV. Analytical Model of the Buffer Memory of an OpenFlow Switch in a Software-Defined Network (SDN). *Advanced Engineering Research (Rostov-on-Don)*. 2026;26(2):2221. https://doi.org/10.23947/2687-1653-2026-26-2-2221

Оригинальное теоретическое исследование

Аналитическая модель буферной памяти OpenFlow коммутатора программно управляемой сети SDN

Г.В. Слюсарев, В.П. Мочалов, Д.В. Гостева ✉
Северо-Кавказский федеральный университет, г. Ставрополь, Российская Федерация
✉ DVGus@yandex.ru

Аннотация

Введение. Достоверное выявление вероятностно-временных особенностей узлов коммутации необходимо для оценки количественных характеристик программно конфигурируемых сетей. Широко используемые методы теории массового обслуживания (ТМО) лишь приближенно определяют и моделируют процессы в коммутаторе OpenFlow и его буферной памяти. Это ведет к получению заниженных, не реалистичных характеристик проектируемых сетевых устройств, перегрузке буферной памяти коммутаторов и потере сетевых пакетов. Проблему может решить иной подход к моделированию. Цель представленной работы — создание и исследование аналитической модели буферной памяти OpenFlow коммутатора программно управляемой сети SDN с использованием усовершенствованных методов ТМО.

Материалы и методы. Задействовали аппарат дискретного преобразования Лапласа-Стилтьеса. Учитывались статистические характеристики потоков пакетов и пропускная способность каналов связи при заданной вероятности потерь пакетов. Модель буферной памяти коммутатора OpenFlow базируется на математическом аппарате ТМО. Его строили в предположении рекуррентности входных потоков данных с групповым поступлением. Исходили из схематически представленных структур коммутатора, набора его записей и графового описания передачи выходящих из коммутатора сетевых пакетов. Приняли как допустимые два ограничения модели:

– произвольное распределение зависимости между объемом потоков данных и временем их обслуживания;

– дискретность распределения структуры информационных потоков.

Результаты исследования. Созданная модель интегрирует вероятность потери потоков пакетов, их статистические характеристики, пропускную способность вычислительных устройств и процедуру многофазного обслуживания. При проверке работоспособности модели приняли, что загрузка коммутатора увеличивается от 0,1 до 0,9, а вероятность потери — от 10^{-3} до 10^{-6}. Для этих показателей выяснили, как от загрузки коммутатора зависят объем буферной памяти и время ожидания. В первом случае минимальное значение (объем памяти) — 0,201, максимальное — 10564. Во втором — 0 и 470 мс соответственно. Для имитационного моделирования минимум по времени — 0 мс, максимум — 2300 мс. Показатели имитационного и аналитического моделирования близки при загрузке менее 50 % и увеличиваются в несколько раз при загрузке более 50 %. Показатели резко возрастают с загрузкой до 70 %, а затем кратно увеличиваются.

Обсуждение. При невысокой сетевой загрузке очереди не переполняются, пакеты не теряются, сохраняются линейные зависимости. При средней и высокой загрузке обработка потоков пакетов описывается нелинейными зависимостями. Результаты аналитического и имитационного моделирования расходятся из-за взрывообразного характера самоподобного сетевого трафика и его приближенного описания распределением Парето. Загруженность коммутатора определяет целесообразность применения предложенного подхода. Модель подойдет при проектировании элементов программно конфигурируемых сетей для анализа их устойчивости при различных информационных воздействиях.

Заключение. Предложенная аналитическая модель SDN определяет значения и дисперсии объема буферной памяти коммутатора, а также объемы памяти для построения таблиц потоков адресации. Работоспособность решения проверили при загрузке коммутатора от 0,1 до 0,9. Планируется создать модель, учитывающую потоки заявок как из внешней сети, так и от сервера.

Ключевые слова: совершенствование методов теории массового обслуживания, аналитическая модель буферной памяти OpenFlow, потеря сетевых пакетов, имитационное моделирование коммутатора

Благодарности. Авторы выражают благодарность редакции, академическому редактору и рецензентам за внимательное отношение к статье и замечания, которые позволили повысить ее качество.

Для цитирования. Слюсарев Г.В., Мочалов В.П., Гостева Д.В. Аналитическая модель буферной памяти OpenFlow коммутатора программно управляемой сети SDN. *Advanced Engineering Research* (*Rostov-on-Don*). 2026;26(2):2221. https://doi.org/10.23947/2687-1653-2026-26-2-2221

Introduction. Programmable SDN networks and the OpenFlow network device management protocol were created to improve automation of modern computer network management and support the specified quality of cloud services, that is, for tasks solved by dynamically redistributing network resources between users. One of the core elements of software-defined networks is the OpenFlow flow switch, and its most important component is the buffer memory. Using an internal software pipeline, the switch distributes and balances loads between network links. This provides the flexibility of centralized network resource management.

Quantitative performance evaluation of networks involves identifying the probabilistic and temporal characteristics of switching nodes. This is done by widely used queuing theory (QT) methods, which only approximately define and model processes in an OpenFlow switch and its buffer memory. This results in underestimated, unrealistic performance of the designed network devices, overloading the switch buffer memory, and loss of network packets. To address this challenge, a different modeling approach is proposed. This work aims to develop and evaluate an analytical model for OpenFlow switch buffer memory in SDN using advanced queueing theory approaches. For this, the discrete Laplace-Stieltjes transform (DLT) is used, taking into account the statistical characteristics of packet flows and the throughput of communication channels for a given packet loss probability. A similar mathematical model based on the classical QT is described in [1], but it does not eliminate the problems that arise due to the approximate description of network processes.

Scientific literature offers design solutions for improving data network elements, demonstrating the importance and relevance of the scientific and technical problem being solved. OpenFlow models for switches in software-defined networks (SDN) are the subject of works by Raghav S.S., Baskakov A.E., Volkov A.S., Filippov I.A., Nikishin K.I., Gurin E.I., Tikhonenko O.M., Gorbunov A.V., Lebedev A.V., Samouylov K.E.., Shalimov I.A., and others. In [2], a fairly comprehensive review of network solutions based primarily on classical QT is given. Finite-capacity systems with a recurrent input flow, a Markovian service process, and infinite-capacity buffers are studied. The modeling methods are based on replacing recurrent arrival flows with simpler ones. The dependence of the volume of requests on the service time is assumed to be given or neglected. The probability of packet loss is supposed to be negligible. Such models often assume packet loss when the buffer is fully occupied, but this assumption is not always accurate. They also assume an unlimited total memory capacity and a well-defined relationship between message volume and service time. Most publications lack finite expressions for analyzing switch service quality, taking into account the specifics of buffer memory allocation, loss indicators, and load. In these cases, the authors use numerical methods from off-the-shelf software packages.

The scientific novelty of this research lies in the improvement of classical queuing theory methods, which support the analysis of systems with arbitrarily distributed arrival and service requests. The practical significance of the results is confirmed by two factors. First, an analytical model of the buffer memory of an OpenFlow switch in an SDN network is created, allowing for a sufficiently accurate analysis. Second, it opens the possibility of selecting alternative design options at the design stage. The efficiency of the proposed model is assessed under simulation. The resulting performance metrics provide evaluating the efficiency of an OpenFlow switch in an SDN network.

Materials and Methods. The incoming SDN network packet flow enters an OpenFlow switch, which extracts metadata, checks it against entries in address tables, and determines the forwarding direction. The network operating system of the SDN controller distributes the packets into flows configured by the OpenFlow classifier. The OpenFlow switch integrates multiple flow tables managed by the central controller through instructions and packet forwarding. The SDN controller periodically updates its records of the network structure, load, resources used, and reserves. Accordingly, it establishes forwarding rules for all incoming flows, sequentially distributes them to the switch OpenFlow output ports, sets new action parameters, and performs distribution or transmission back into the network. This utilizes the switch software pipeline, which consists of sequential flow tables.

Packets not identified in the flow tables are sent to another switch port or forwarded to the controller for field modification. The SDN controller generates switch output ports, address tables, and data flow classifiers [3].

Much attention is paid to issues of network device management, and ready-made design solutions are available. Despite this, new, efficient methods and protocols for network load management are being developed, and network management protocols are constantly being refined. The short period of operational experience with SDN networks and the lack of systematic testing results for switching equipment on the OpenFlow platform necessitate further research using analytical and simulation models.

Thus, the switch structural diagram can be represented as a queueing system (QS) with limited memory. Its input receives a flow of packets with arbitrary distributions of the number and volume of requests. The operating mode of such a system will be stationary if the average number of requests arriving per unit of time does not exceed the maximum possible service rate. One of the most challenging tasks is determining the mean values and variances of the switch total buffer memory capacity, as well as the memory capacity for constructing flow tables [4]. Assume that there are no packet flow queues at the input of a high-performance central control server. In this case, the switch operation in steady-state mode can be described by the apparatus of queuing systems with Laplace-Stieltjes transforms. Data distribution and processing using a software pipeline is modeled as a multiphase random process [5].

The advanced QT method used in this paper allows for the creation of models of numerous real systems, including switching ones. It can be reasonably assumed that such systems have stationary characteristics, and that the parameters of the input packet flow do not affect the order and time of its processing. We also assume that each flow is characterized by a random number of packets, that is, its volume takes only positive discrete values, and this, with a high degree of probability, reflects the real situation.

A switch forwarding pipeline consists of one or more address tables connected in series. Input network traffic, described by an arbitrary distribution law, enters the conveyor and is sequentially processed by k address tables.

It is assumed that the query processing time for each table is random and exponential. The sum of the times of all processing stages forms the final service time distribution (the Erlang k distribution). It is known that a k-order Erlang flow with parameters (μ, r) can be represented as the sum of r random flows with parameters μ_i, and the characteristics of the system under consideration can be obtained from the characteristics of its constituent elements [6]. Then, the stationary probabilities $p(i_1,\ldots, i_r)$ of the system under consideration:

$$p\left(i_1,\ldots,i_r\right)=\prod_{k=1}^{r}\lim_{t\to\infty} p\left\{i_k\left(t\right)=i_k\right\}.$$

The structure of the switch record set is shown in Figure 1.

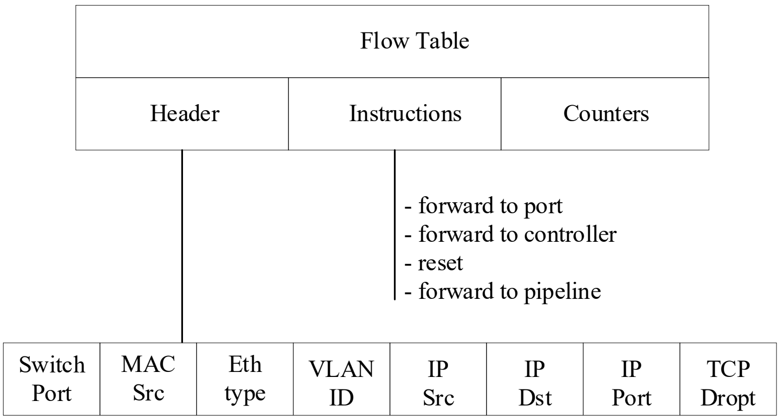

Fig. 1. Switch entry set structure

A simplified structure of an OpenFlow switch is shown in Figure 2.

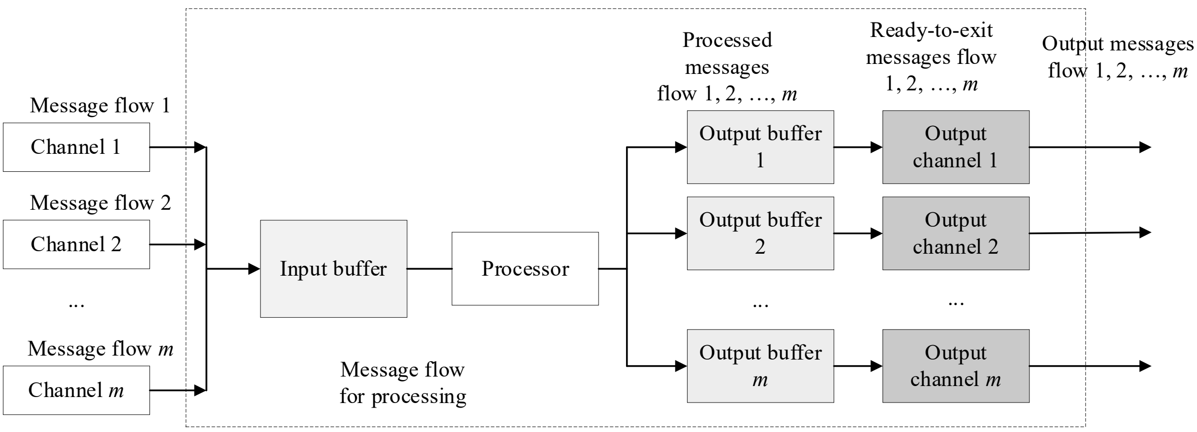

Fig. 2. OpenFlow switch structure

The time intervals between packets are random. The input flow is Markovian, but its service time does not obey an exponential distribution, that is, the analyzed processing of input flows contains a Markovian component. Therefore, this system can be studied using the mathematical apparatus of semi-Markovian processes [7], and the distribution function of the time intervals of the process in state *i* will be an arbitrarily distributed random variable [8]. The intervals between the end of service and the next arrival of service requests are subject to the same distribution. We also assume that random packet flows, determined by the Erlang distribution function, arrive via *m* channels with intensities $a_i \left(i=\overline{1,m}\right),\ \lambda=\sum_{i=1}^{m} a_i$.

For $M/G/1/n$ system, the Laplace-Stieltjes transform of the distribution function of the volume of the serviced packet $R(x)$:

$$l_i(s)=1-\frac{a_i}{g_i}\left[\frac{1}{f}-\frac{f}{(s+f)^2}\right].$$

Here, $g_i > 0$ — packet reception delay time on the *i*-th channel, f — distribution parameter, s — complex parameter of the Laplace-Stieltjes transform. In this expression, it functions as a multiplier in the exponent and allows finding the moments of the distribution of a random variable of any degree.

Mean value and variance:

$$l_{1i}=\frac{2a_i}{g_i f^2}=\frac{2\rho_i}{f},\ l_{2i}=\frac{2a_i}{g_i f^3}=\frac{6\rho_i}{f^2},$$

where ρ_i — input channel loading, $\left(l_{2i}-l_{1i}{}^2\right)=\frac{2\rho_i}{f^2}(3-2\rho_i)$.

Then the volume of packet flow is DLT:

$$\delta(s)=\prod_{i=1}^{m}\left\{1-\frac{a_i}{g_i}\left[\frac{1}{f}+\frac{f}{(s+f)^2}\right]\right\}.$$

Mean values and variance:

$$\delta_1=\frac{2}{f}\sum_{i=1}^{m}\rho_i,\ \left(\delta_2-\delta_1^2\right)=\frac{2}{f^2}\sum_{i=1}^{m}\rho_i\left(3-2\rho_i\right).$$

We consider that the probability of packet flow loss [9]:

$$\rho_{\Pi}=1-R(V),$$

where $R(V)=\int_o^V D(V-x)\,dL(x)$, $L(x) = 1 - e^{-fx}$ — distribution function (DF) of the packet flow volume; $D(x) = p(\delta < x)$ — DF of the total volume of packet flow δ.

DF moments $R(x)$:

$$r_1=\delta_1+\varphi_1,\quad r_2=\varphi_2+2\varphi_1\delta_1+\delta_2,$$

where $\varphi_1 = 1/f$ — mean packet flow volume; δ_1, δ_2 — moments of the total volume of packet flow.

Finding the explicit form of the DF $D(x)$ does not seem possible. In [10], it is shown that when performing calculations, this function can be approximated quite accurately by the expression:

$$D(V)=p_0+(1-p_0)\frac{\gamma(p,gx)}{\Gamma(p)}.$$

Here, $\gamma(p,gx)=\int_0^{gx} t^{p-1}e^{-t}dt$ — incomplete gamma function; p_0 — probability of no service requests; $\Gamma(p) = \gamma(p, \infty)$ — gamma function; p and g — parameters determined from the condition of equality of the corresponding moments of the switch memory volume:

$$p=\frac{r_1^2}{r_2-r_1^2},\quad g=\frac{r_1}{r_2-r_1^2}.$$

The numerical values of the DF $D(x)$ are obtained through simulation modeling by standard numerical methods, which allow the solution to be reduced to a finite number of arithmetic operations. The results of the simulation modeling are virtually identical to the results of the analytical modeling described in this article, demonstrating the validity of the developed model.

When forming network flows, the following actions are performed [11].

1. Identification of the traffic packet flow.
2. Packet-to-flow classification for the flow being formed.
3. Formation of network packet flows or transmission of an unidentified packet to the management controller.

The function of allocating the service time for the input packet flow $B(t) = p + (1 - p)(1 - e^{pt})$, and its DLT [12]:

$$\beta(q) = p + \frac{(1-p)p}{p+g} = \frac{p(1+q)}{p+g}.$$

Mean service time:

$$\beta_1 = -\beta'(0) = \frac{1-p}{p}.$$

If the system load $\rho = a\beta_1 = a(1 - p)/p$, then the DLT service waiting time:

$$W(q) = \frac{(1-\rho)(p+q)}{p+q-a(1-p)} = \frac{(1-p)(p+q)}{q+p(1-\rho)}.$$

Then the mean value of the waiting time is: $W_1 = -W'(0) = \dfrac{p}{p(1-\rho)}$.

The image has the form of rational fraction $(A_n(p))/(B_n(p))$. $P_1, P_2, \ldots, P_n$ — roots of multiplicity $r_1, r_2, \ldots, r_n$, where $r_1 + r_2 + \cdots + r_n = m$ и $B_m(p) = \beta_0 (p - P)_1^{r_1} (p - P_2)^{r_2} \ldots (p - P_n)^{r_n}$ .This means that the original can be found from formula [13]:

$$f(t) = \sum Res\left[\frac{A_n(p)e^{pt}}{B_m(p)}\right].$$

For simple roots of the denominator $P_1, P_2, \ldots, P_n$:

$$f(t) = \sum \frac{A_n(P_k)}{B_m(P_k)} e^{p} k^{t}.$$

The inversion of the DLT function $W(q)$ is determined by the relation:

$$W(t) = \sum Res\left[\frac{(1-\rho)(\rho+q)}{q(q+p(1-\rho))} e^{qt}\right].$$

The distribution function of random variable V has the form:

$$V(t) = p\{V < t\} = \int_0^t W(t-u)\,dU = 1 - e^{-(1-\rho)\mu t}.$$

For the case $p\{W > 0\} = 1 - W(0) = \dfrac{(n\rho)^n p_0}{n!(1-\rho)}$, mean value of stationary waiting time:

$$W_1 = EW = \int_0^\infty dW(t) = \frac{n^{n-2}\rho^n p_0}{\mu(1-\rho)^2 (n-1)!}.$$

Mean value of stationary service time:

$$V_1 = EV = \int_0^\infty t\,dV(t) = \beta_1 + T_1 = \frac{1}{\mu} + \frac{n^{n-2}\rho^n p_0}{\mu(1-\rho)^2 (n-1)!}.$$

These random variables are approximated by the DF $Z(x) = p_0 + (1-p_0)\dfrac{\gamma(p,gx)}{\Gamma(p)}$. Here, $\gamma(p,gx) = \int_0^{gx} t^{p-1}e^{-t}dt$, $\Gamma(p) = \gamma(p,\infty)$, p_0 — stationary probability of no requests:

$$p = \frac{\delta_1^2}{(1-p_0)\delta_2 - \delta_1^2}, \quad g = \frac{(1-p_0)\delta_1}{(1-p_0)\delta_2 - \delta_1^2}.$$

Obviously, the phases of transmission of network packets leaving the switch can be described by the graph (Fig. 3) [14].

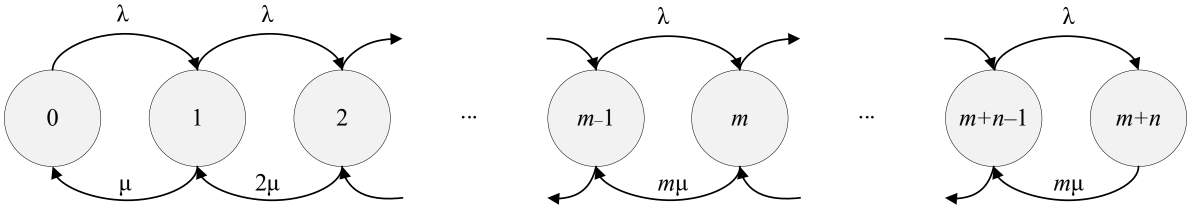

Fig. 3. QS state and transition graph *M*/*M*/*m*/*n*

From [15], the probability of system *M*/*M*/*m*/*n* being in the state p_k is known:

$$p_k = \frac{\dfrac{\rho^k}{k!}}{\sum_{k=0}^{m}\dfrac{\rho^k}{k!} + \dfrac{\rho^{m+1}}{mm!} - \dfrac{1-\left(\dfrac{\rho}{m}\right)^n}{1-\dfrac{\rho}{m}}},\ 0 \le k \le m.$$

Here, ρ = λ/μ — load.

Accordingly, the probability of the state p_{m+s} is:

$$p_{m+s} = \frac{\dfrac{\rho^m}{m\,!}\left(\dfrac{\rho}{m}\right)^s}{\sum_{k=0}^{m}\dfrac{\rho^k}{k!} + \dfrac{\rho^{m+1}}{mm!} - \dfrac{1-\left(\dfrac{\rho}{m}\right)^n}{1-\dfrac{\rho}{m}}},\ 0 \le \rho \le n.$$

Then the intensity of the output flow is: $y = \lambda - \lambda p_{m+n} = \lambda \sum_{i=0}^{m+-1} p_i$.

$$p_0 = \frac{\rho}{\sum_{k=0}^{m}\dfrac{\rho^k}{k!} + \dfrac{\rho^{m+1}}{mm!} - \dfrac{1-\left(\dfrac{\rho}{m}\right)^n}{1-\dfrac{\rho}{m}}}.$$

$$p_{m+n} = \frac{\dfrac{\rho^m}{m!}\left(\dfrac{\rho}{m}\right)^n}{\sum_{k=0}^{m}\dfrac{\rho^k}{k!} + \dfrac{\rho^{m+1}}{mm!} - \dfrac{1-\left(\dfrac{\rho}{m}\right)^n}{1-\dfrac{\rho}{m}}} = \frac{\dfrac{\rho^m}{m!}\left(\dfrac{\rho}{m}\right)^n}{\rho} p_0.$$

According to [16], taking into account lost packets, the output flow intensity:

$$y = \lambda\left(1 - p_{m+n}\right) = \lambda\left(1 - \frac{\dfrac{\rho^m}{m!}\left(\dfrac{\rho}{m}\right)^n}{\rho} p_0\right) = \lambda\frac{\rho - \dfrac{\rho^m}{m!}\left(\dfrac{\rho}{m}\right)^n}{\rho} p_0.$$

Time spent by packets in the system:

$$V = \frac{N}{m\mu\left(1-p_0\right)} = \frac{\sum_{k=0}^{m+n} kp_k}{m\mu\left(1-p_0\right)}.$$

Waiting time for packets in the queue:

$$W = \frac{N_0}{m\mu\left(1-p_0\right)} = \frac{\sum_{k=m+1}^{m+n}\left(k-m\right)p_k}{m\mu\left(1-p_0\right)}.$$

Service time:

$$T_{об} = V - W = \frac{\sum_{k=0}^{m+n} k p_k}{m\mu(1-p_0)} - \frac{\sum_{k=m+1}^{m+n} (k-m) p_k}{m\mu(1-p_0)} = \frac{1}{m\mu} + \frac{(m-1)\sum_{k=m}^{m+n} p_k + \sum_{k=2}^{m-1} (k-1) p_k}{m\mu(1-p_0)}.$$

The presented expressions allow deriving the basic characteristics of an OpenFlow switch. In this case, packet flow processing and management, as well as interaction with the central network controller are taken into account. Buffer memory is considered a shared, dynamically allocated resource and is determined by constraints on the percentage of lost packets for a given loss probability. Acceptable model constraints:

– arbitrary distribution of the relationship between the volume of data flows and their service time;

– assumed discreteness of the distribution of the structure of information flows.

Research Results. Thus, as part of this study, a model of an OpenFlow switch is developed to calculate and evaluate its buffer memory. Using the mathematical apparatus of semi-Markovian processes and Laplace-Stieltjes transforms, the following parameters are defined:

– packet flow loss probability;

– their statistical characteristics;

– computing device throughput.

Packet flow processing consists of multiple independent sequential phases, and the service time across them is exponentially distributed. Therefore, the proposed model is based on a multiphase servicing procedure, in which the service time is determined by the n-th order Erlang distribution, and the moments of the distribution functions are found using the Laplace-Stieltjes transform.

Tables 1 and 2 show the results of analytical modeling of the switch buffer memory volume and the dependence of the mean service waiting time on the load. In this case, it increases from 0.1 to 0.9. Switch memory capacity is measured by the number of mean input packet flows.

Table 1

Dependence of Buffer Memory Size on Switch Load

Probability of loss, p_n	Loading switch buffer memory, ρ								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
10^{-3}	0.201	0.307	0.412	0.478	0.617	37.05	69.07	94.36	178.87
10^{-4}	0.403	0.481	0.521	0.680	2.15	4.31	21.05	232.7	831.3
10^{-5}	0.762	0.790	0.840	0.932	3.76	7.83	16.07	476.5	983.2
10^{-6}	0.937	0.951	1.511	1.79	5.07	16.08	87.13	748.1	10564

Table 2

Dependence of Mean Waiting Time on Switch Load, ms

Probability of loss, p_n	Loading switch buffer memory, ρ								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
10^{-4}	0.00	0.00	0.00	0.02	0.07	0.17	30	120	470
10^{-4}	0.00*	0.00*	0.01*	0.04*	0.12*	0.53*	90*	510*	2300*

Note: * Results of switch simulation are marked.

The simulation results were obtained using self-similar network traffic described by a Pareto distribution with a parameter of 1.5. If the system load is less than 50%, the simulation-based characteristics for the $P/G/1$ system deviate slightly from the mean values obtained in the analytical modeling. At loads exceeding 50%, the values increase several-fold. The indicators increase sharply as the switch load reaches 70% and increase exponentially if this level is exceeded.

Discussion. The results of this scientific work were obtained under the condition that the distribution function of the intensity of the packet flow was independent of the phase that processed them, as well as the constant nature of the volume of the packet flow. Only in this case the result of the Laplace–Stieltjes transformation was valid for the memory size of the switch.

The main drawback of the results is that the obtained estimates are only reliable at the level of partial distributions. The OpenFlow network switch model is built on the assumption of using shared dynamically allocated buffer memory, and the considered phases of request flow processing are assumed to be independent. Therefore, the results of the study can only be used if the time is equal to:

– input of a packet flow into the switch memory;

– its output to the communication channel.

Such conditions should be considered acceptable, taking into account the capabilities of modern data processing and transmission tools.

The simulation results presented in Tables 1 and 2 are obtained taking into account the specified loss characteristics and current network load. At low loads, queues do not overflow, packets are not lost, and linear dependences are maintained. At moderate, and specifically high loads, network packet flow processing is described by nonlinear dependences. The discrepancy between the analytical and simulation results is due to the explosive nature of self-similar network traffic and the rather approximate nature of its description by the Pareto distribution.

Obviously, the feasibility of using the proposed model is determined by the current switch load. This model can be used in the design of network elements of software-defined networks to analyze their resilience to various information impacts.

The results of the presented research for a load of up to 50% are practically identical to the results obtained within the framework of classical methods of queuing theory [5]. Critical losses of input packet flows start at a switch load of 50%. A further increase in load is not considered; therefore, neither the behavior nor the stability of the system under higher loads are assessed.

The performance level of the proposed model corresponds to the known results of simulation modeling using self-similar network traffic described by the Pareto distribution.

Conclusion

1. An analytical model of the OpenFlow buffer memory of a software-defined network (SDN) switch was developed to determine:

– mean values and variance of the switch total buffer memory;

– memory volumes for constructing address flow tables.

2. Analytical expressions were obtained to determine the following characteristics of the quality of service of a software-defined network switch:

– output packet flow rate, taking into account losses;

– packet sojourn time in the system;

– packet waiting time in the queue;

– service time;

– dependence of the buffer memory size and the average service waiting time on the load.

3. The model performance was confirmed by a comparison of the results of analytical and simulation modeling when the switch load varied from 0.1 to 0.9.

4. The model allows testing SDN networks without using physical equipment to justify switch parameters taking into account its environment.

It is expected that the next article, devoted to the development of a model for the functioning of an SDN switch, will take into account the flow of requests both from the external network and from the management controller (server).

References

1. Samouylov KE, Shalimov IA, Buzhin IG, Mironov YuB. Model of Functioning of Telecommunication Equipment for Software-Configurated Networks. *Modern Information Technologies and IT-Education*. 2018;14(1):13–26. https://doi.org/10.25559/SITITO.14.201801.013-026

2. Ivanova D, Adou Y, Markova E, Gaidamaka Yu, Samouylov K. Mathematical Framework for Mixed Reservation- and Priority-Based Traffic Coexistence in 5G NR Systems. *Mathematics.* 2023;11(4):1046. https://doi.org/10.3390/math11041046

3. Baskakov AE, Volkov AS. Resources Managing Algorithm for Transport Software-Defined Communication Network *Trudy MAI*. 2020;(115). https://doi.org/10.34759/trd-2020-115-06

4. Stepanov MS, Stepanov SN, Kanischeva MG, Kroshin FS. Analysis of Procedures to Ensure the Required QoS Indicators in Multiservice Access Nodes. In: *Proceedings of the XXVI International Conference. on "Distributed Computer and Communication Networks: Control, Computation, Communications (DCCN)"*. Moscow: V.A. Trapeznikov Institute of Control Sciences, Russian Academy of Sciences; 2023. Iss. 1. P. 47–55. https://doi.org/10.25728/dccn.2023.007

5. Mochalov VP, Linets GI, Bratchenko NY, Govorova SV. An Analytical Model of a Corporate Software-Controlled Network Switch. *Scalable Computing: Practice and Experience*. 2020;21(2):337–346. https://doi.org/10.12694/scpe.v21i2.1698

6. Malakhov SV, Yakupov DO, Osipova AA, Kopylova DA, Zelenina EA. The Use of a Queuing System to Study the Characteristics of the Communication Channel in IoT Networks. *Vestnik NSU. Series: Information Technologies*. 2024;22(1):49–61. https://doi.org/10.25205/1818-7900-2024-22-1-49-61

7. Tikhonenko OM. Generalized Erlang Problem for Service Systems with Finite Total Capacity. *Communication Network Theory*. 2005;41:243–253. https://doi.org/10.1007/s11122-005-0029-z

8. Antonova PV. Principles of Developing a Limited-Queue Mass Service System on the .NET Platform. *Software Systems and Computational Methods*. 2023;(2):15–28. https://doi.org/10.7256/2454-0714.2023.2.43403

9. Voronova AG. Typification of Projects for the Transition to Cloud Services. *Advanced Engineering Research* (*Rostov-on-Don*). 2024;24(3):274–282. https://doi.org/10.23947/2687-1653-2024-24-3-274-282

10. Gorbunova AV, Lebedev AB. On Estimating the Characteristics of a Fork-Join Queueing System with Poisson Input and Exponential Service Times. *Advances in Systems Science and Applications*. 2023;23(2):99–114. https://doi.org/10.25728/assa.2023.23.2.1351

11. Mochalov VP, Bratchenko NYu, Moiseenko VA, Gosteva DV. *A Simulation Model of Algorithms for the Operation of Nodes of Future Communication Networks*. Certificate of Software State Registration No. 2019618220, 2019. (In Russ.) URL: https://new.fips.ru/registers-docview/fips_servlet?DB=EVM&DocNumber=2019618220&-TypeFile=html (accessed: 09.06.2026.)

12. Topilin IV, Han M, Feofilova AA, Beskopylny NA. Comparative Analysis of Neural Network and Machine Learning Models for Short-Term Traffic Flow Prediction on Shenzhen Expressway. *Advanced Engineering Research* (*Rostov-on-Don*). 2025;25(4):350–362. https://doi.org/10.23947/2687-1653-2025-25-4-2215

13. Gorbunova AV. On the Features of Service Speed Control in Fork-Join Systems with Pareto Distribution of Service Time. *Proceedings of Voronezh State University. Series: Systems Analysis and Information Technologies*. 2025;(4):53–62. https://doi.org/10.17308/sait/1995-5499/2024/4/53-62

14. Dudin SA, Dudin AN, Dudina OS, Chakravarthy SR. Analysis of a Tandem Queuing System with Blocking and Group Service in the Second Node. *International Journal of Systems Science: Operations and Logistics*. 2023;10(1):2235270. https://doi.org/10.1080/23302674.2023.2235270

15. Samoylenko VV. Concept of a Multilevel Network Infrastructure for Monitoring Agricultural Facilities Based on Wireless Sensor Networks. *Advanced Engineering Research* (*Rostov-on-Don*). 2025;25(4):371–382. https://doi.org/10.23947/2687-1653-2025-25-4-2238

16. Gorbunova AV, Lebedev AV. On the Features of Service Rate Control in Fork-Join Queueing System. *Automation and Remote Control*. 2024;12:70–88. https://doi.org/10.31857/S0005231024120048

About the Authors:

Gennady V. Slyusarev, Dr.Sci. (Eng.), Leading Researcher of the Department of Science, Professor of the Department of Civil Engineering and Prototyping, Institute of Advanced Engineering, North-Caucasus Federal University (2, Kulakov Pr., Stavropol, 355029, Russian Federation), SPIN-code, ORCID, ScopusID, ResearchGate, gsliusarev@ncfu.ru

Valery P. Mochalov, Dr.Sci. (Eng.), Professor of the Department of Digital, Robotic Systems and Electronics, Institute of Advanced Engineering, North-Caucasus Federal University (2, Kulakov Pr., Stavropol, 355029, Russian Federation), SPIN-code, ORCID, ScopusID, ResearcherID, ResearchGate, mochalov.valery2015@yandex.ru

Dar'ya V. Gosteva, programmer, North-Caucasus Federal University (2, Kulakov Pr., Stavropol, 355029, Russian Federation), SPIN-code, ORCID, ScopusID, ResearcherID, ResearchGate, DVGus@yandex.ru

***Claimed Contributorship*:**
GV Slyusarev: conceptualization.
VP Mochalov: project administration, writing – review & editing.
DV Gosteva: software.

***Conflict of Interest Statement:* the authors declare no conflict of interest.**

All authors have read and approved the final manuscript.

Об авторах:

Слюсарев Геннадий Васильевич, доктор технических наук, ведущий научный сотрудник департамента науки, профессор департамента «Строительная инженерия и прототипирование» института перспективной инженерии Северо-Кавказского федерального университета (355029, Российская Федерация, г. Ставрополь, пр. Кулакова, 2), SPIN-код, ORCID, ScopusID, ResearchGate, gsliusarev@ncfu.ru

Мочалов Валерий Петрович, доктор технических наук, профессор департамента «Цифровые, робототехнические системы и электроника» института перспективной инженерии Северо-Кавказского федерального университета (355029, Российская Федерация, г. Ставрополь, пр. Кулакова, 2), SPIN-код, ORCID, ScopusID, ResearcherID, ResearchGate, mochalov.valery2015@yandex.ru

Гостева Дарья Валерьевна, программист, Северо-Кавказский федеральный университет (355029, Российская Федерация, г. Ставрополь, проспект Кулакова, 2), SPIN-код, ORCID, ScopusID, ResearcherID, ResearchGate, DVGus@yandex.ru

Заявленный вклад авторов:

Г.В. Слюсарев: разработка концепции.

В.П. Мочалов: административное руководство исследовательским проектом, написание рукописи – внесение замечаний и исправлений.

Д.В. Гостева: разработка программного обеспечения.

Конфликт интересов: **авторы заявляют об отсутствии конфликта интересов.**

Все авторы прочитали и одобрили окончательный вариант рукописи.

Received / Поступила в редакцию 26.02.2026
Reviewed / Поступила после рецензирования 01.04.2026
Accepted / Принята к публикации 17.05.2026