Research on Network Flow Based on Statistical Analysis Methods

Maruf Juraev[1], Inomjon Yarashov[2], Adilbay Kudaybergenov[3], Alimdzhan Babadzhanov[4], Zilolaxon Mamatova[5]
Tashkent Institute of Technology, Management and Communication, Tashkent, Uzbekistan[1]
Jizzakh branch of the National University of Uzbekistan Named after Mirzo Ulugbek, Jizzakh, Uzbekistan[1]
Tashkent International University of Education, Department of Information Technology, Tashkent, Uzbekistan[2]
University of World Economy and Diplomacy, Tashkent, Uzbekistan[2]
Karakalpak State University, Nukus, Uzbekistan[3]
National University of Uzbekistan Named After Mirzo Ulugbek, Tashkent, Uzbekistan[3]
Engineering Federation of Uzbekistan, Tashkent, Uzbekistan[4]
Fergana State University, Fergana, Uzbekistan[5]

***Abstract*—To solve the problem of detecting unexpected situations in network traffic, it is proposed to determine the normal, unexpected states and behavior of the system using statistical methods. The statistical analysis methods used are the mean, variance, asymmetry coefficient, kurtosis coefficient, opposition coefficient and entropy coefficient. All statistical analysis methods allow for a deeper understanding of the uniqueness of the data. Each coefficient provides different measurement values, with which it is possible to determine the general characteristics of the data. The application of statistical analysis methods in network traffic is one of the most common methods for implementing the technology of detecting unexpected situations. For this, network traffic recorded in real time in laboratory conditions is used. Packet features of the network traffic are extracted from the recorded data, and then statistical analysis is performed using the extracted features. Markov chain is one of the most effective tools for analyzing situations and events occurring in network traffic. Markov chain is formed using the results of statistical analysis, and a Markov chain is constructed. This approach represents the state of the network flow in a probabilistic model and serves as an effective tool in monitoring the network flow.**

***Keywords*—Unexpected situation; mean; variance; asymmetry coefficient; kurtosis coefficient; opposition coefficient; entropy coefficient; packet analysis; markov chain; packet features**

I. Introduction

Today, ensuring information protection is one of the most pressing issues. The increase in the volume of data also creates new problems. This means that network security officers need to strengthen measures to eliminate or prevent problems in the proper organization of network management [32]. Traditional methods do not provide sufficient results in the face of new attacks [7]. Therefore, it is necessary to use modern analysis methods. In particular, traffic analysis, anomaly detection [13], and deep analysis of packet features are effective tools [15]. These methods are widely used by many researchers due to the fact that they can be implemented in real time [16].

Analysis through packet features is a time-saving method for detecting unexpected events in network operations [33]. Researchers use neural network, machine learning [6], deep learning and analytical analysis methods [31] to analyze packet features [3]. Statistical analysis methods have been little studied. Therefore, statistical analysis methods were used in this research work.

The research study investigated unexpected network traffic patterns. To do this, first, the network traffic was recorded, and packet features were extracted. The data was recorded in a laboratory environment using the Wireshark program, and packet features were extracted based on the developed algorithm.

Statistical analysis methods [28], including mean, variance, asymmetry coefficient, kurtosis coefficient, opposition coefficient and entropy coefficient, were used to analyze the extracted packet features [1], [2]. Markov chain is the most effective method for in-depth analysis of unexpected situations [21]. As a result of the research work, statistical analysis methods and Markov chain architecture were developed to identify unexpected situations in network flow [14]. Based on the results obtained, an assessment was made of unexpected situations in network flow.

The novelty of the study is that it proposes a Markov chain model, combining statistical indicators based on real traffic data. This allows not only to identify unexpected situations in the network, but also to predict their development trends in advance.

The study consists of six sections: Introduction, Related works, Problem statement, Proposed methodology, Experimental results and discussions, and Conclusion.

II. Related Works

Ramachandran et al. [1] have studied probability theory and mathematical analysis in detail in statistical analysis. In their book, they have justified the analysis of theoretical knowledge and practical skills necessary for solving real-life statistical problems [26].

Hu et al. [2], performed quantitative data analysis and sample variance analysis. Based on the experiments, they proposed new solutions for variance analysis [35].

Alasmar and Zakhleniuk used variance from statistical analysis algorithms to extract link data from the incoming data.

The proposed link size estimation approach measures the statistical parameters and then measures the rate of change in traffic. The heavy length distributions performed better than the Gaussian model [3].

Chydzinski [4] studies the effect of interdependence of packet arrival time and service delivery on the degradation of queuing system performance in network operation.

Kasse et al. [5] present an approach using the Path-scan model to identify paths that exhibit anomalous behavior in network communications. They developed a Markov model consisting of three states to model the states in the network.

Li and Wu used a Markov chain model to calculate the probability of agents forwarding packets in the idle or busy state and analyze the packet forwarding behavior of agents. On this basis, they proposed a game model for packet forwarding strategies [6].

Zhou and Nikaein [7] proposed an adaptive packet collection method to improve the reliability of random access. Random access [30] is not checked for each packet, but instead, random access is checked when the number of collected packets reaches a certain threshold. During packet collection, there may be delays or packet loss. To overcome this problem, a Markov model was used.

Sun and Zhao proposed a Markov chain method for modeling photovoltaic array noise and studied the accuracy of the modeling [8].

Doncel described the average information age of a system [24] using the stochastic hybrid system method. The study investigated the average information age of a queuing system formed by two queues, the first queue storing the system updates and the second one storing the energy required to deliver the state update to the monitor [34]. The queues process one packet [9].

Ott and Swanson study Markov chain models derived from protocols in the TCP (Transmission Control Protocol) paradigm. The researchers study the problems of packet loss, latency, and network congestion [10].

Even studied the convergence rate of Markov chains and optimized functions [23] under the least restrictive assumptions. He showed a theoretical lower bound for methods for sampling stochastic gradients along a Markov chain path and a dependence on the beat time of the Markov chain [11].

Smiesko et al. [12] used statistical parameters to detect DDoS attacks in real time. Statistical analysis methods such as kurtosis coefficient, entropy, correlation coefficient, and Hurst exponent were used to analyze network flow data [17].

Lu uses data transformation analysis to reduce risks in online shopping and ensure online security [34]. Opposition coefficient, correlation coefficient and other methods of statistical analysis were used [13].

Kenyon et al. [14] analyzed the payload of packets using entropy and found that entropy is an effective method for anomaly detection. They compared real-time entropy values with baseline values to detect anomalies in the flow [18].

In calculating the maximum entropy of network traffic, Gu et al. [15] proposed an effective method for detecting behavior-based anomalies, which is commonly applied to TCP and UDP (User Datagram Protocol) traffic data. The calculation used the protocol information and the sending port number features.

Doriguzzi-Corin et al. [16] propose to detect DDoS (Distributed Denial-of-Service) attacks by extracting packet features and exploiting the features.

In the analyzed research works, it was mainly studied on the basis of analysis based on packet data. However, the data in the packets were considered as the basis for analysis, but their interdependence was rarely analyzed. To solve this problem, the research study used a Markov chain in combination with statistical analysis methods, which allows for accurate evaluation of the results obtained.

III. Problem Statement

Developing effective methods for detecting and responding to threats by analyzing large networks has always been relevant [19]. One of the biggest challenges is to detect attacks that exhibit anomalous behavior in selected packet features of network traffic [20]. Detection and response methods must be adaptable to a wide variety of network environments, and these methods should be implemented without human intervention as much as possible [25], [27].

Attack detection should be as accurate as possible. Because a falsely detected attack can lead to a denial of service for legitimate users in network operations. If the detection results in the detection of attacks that are not occurring, the attack will not be detected, meaning the detection process will not work properly [26], [28]. The method used in the detection process should be effective and fast in relation to the various attacks that currently exist.

Research shows that statistical features and statistical processing are a useful approach for detecting anomalous intrusions.

Developing useful methods for detecting unexpected events that occur as a result of device failures or unforeseen events observed in network operations is an important task today. The main requirement for the developed methods is the ability to detect any type of unexpected situation in real time, including data that is unknown in advance and is also scattered over time. For this, the recorded network flow characteristics are used. When recording a network flow, the network flow is recorded in pcap format using the Wireshark program and the flow characteristics are extracted for analysis. The extracted characteristics are stored in csv file format. The flow characteristics are evaluated using statistical analysis methods and Markov chains.

IV. The Proposed Methodology

In the proposed research method, network traffic is recorded, and statistical analysis is performed based on the collected data. Fig. 1 depicts the network traffic data collection and analysis. First, the incoming network traffic data was recorded using the Wireshark program in a laboratory setting. The recorded data is saved in the pacp file format. Second, the data stored in the pcap

file is read based on the developed algorithm, 89 different features are extracted from the read data and saved in the csv file format. This process is carried out in three stages. Reading, feature extraction, and saving the features in the csv file format. The data in the csv file is processed and checked using statistical analysis algorithms [1]. A Markov chain is formed using the results of statistical analysis. The results of statistical analysis are important in determining the set of Markov chain states.

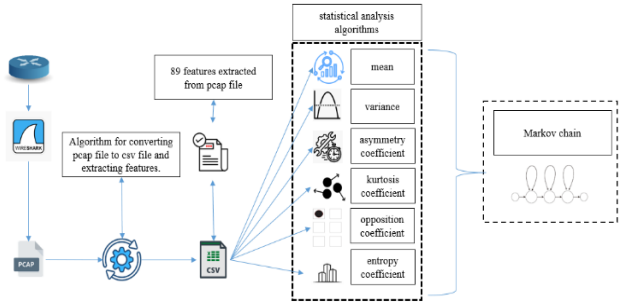

Fig. 1. Statistical analysis structure of network flow feature extraction.

A. Extracting Network Flow Characteristics

Network traffic was captured in a laboratory environment using Wireshark. During the recording, data was transmitted through 6 computers and various network activity conditions were captured. Fig. 1 shows a model of the network traffic capture process. Eighty-nine features were extracted from the traffic data recorded in a laboratory environment. The extracted features allow us to solve the research problem.

In the algorithm in Fig. 2, the data recorded in the pcap file is read through Wireshark and the initial properties of the packet are extracted: *srcIP*, *dstIP*, *srcPort*, *dstPort*, *protocol*, *timestamp*, *headBytes*, *payloadBytes*, *flags*, *windowSize*, *payload*. These properties are the main information for forming the flow data. Based on this information, eighty-nine different properties are determined.

```
1: function nextPacket()
2: packetInfo ← None
3: while pcapPtr < pcapLen do ▷ Packet reading continues
4:    totGen.nextId() ▷ Create a new ID for each package
5:    timeHigh ← pcapData[pcapPtr:pcapPtr + 4] ▷ High timestamp
6:    timeHigh ← struct.unpack(typeI, timeHigh)[0] ▷ Unpacking time
7:    timeLow ← pcapData[pcapPtr + 4:pcapPtr + 8] ▷ Lower timestamp
8:    timeLow ← struct.unpack(typeI, timeLow)[0] ▷ Unpacking time
9:    timeStamp ← 1000000 * timeHigh + timeLow ▷ Timestamp calculation
10:   caplen ← pcapData[pcapPtr + 8:pcapPtr + 12] ▷ Package length
11:   caplen ← struct.unpack(typeI, caplen)[0] ▷ Unpacking the packet length
12:   pcapPtr += 16 ▷ Go to package information
13:   packetData ← pcapData[pcapPtr:pcapPtr + caplen] ▷ Get package information
14:   pcapPtr += caplen ▷ Move the pointer to the next packet.
15:   if isIpv4TCP(packetData) then
16:      packetInfo ← getIpv4Info(packetData, timeStamp) ▷ IPv4 TCP packet
17:      break
18:   else if isIpv6TCP(packetData) then
19:      packetInfo ← getIpv6Info(packetData, timeStamp) ▷ IPv6 TCP packet
20:      break
21:   end if
22: end while
23: return packetInfo ▷ Return packet data
24: end function
```

Fig. 2. Pseudocode for reading packet data from a pcap file.

In Fig. 3, timestamps and flags from each packet of data are used to create a stream from the data. When the stream is complete, the data is written to a csv file.

```
1: function addPacket(packet: BasicPacketInfo)
2:   if packet == None then
3:     return
4:   end if
5:   currentTS ← packet.getTimeStamp() ▷ Packet timestamp
6:   pktFwdFlowId ← packet.getFwdFlowId() ▷ Packet transmission flow identifier
7:   pktBwdFlowId ← packet.getBwdFlowId() ▷ Packet reverse flow identifier
8:   if pktFwdFlowId in self.currentFlows or pktBwdFlowId in self.currentFlows then
9:     if pktFwdFlowId in self.currentFlows then
10:       flowId ← pktFwdFlowId ▷ Flow ID
.    .    .    .    .    .    .    .    .    .    .    .
.    .    .    .    .    .    .    .    .    .    .    .
61: function dumpPayloadToCSV()
62:   open "payload.csv" for writing as csvfile
63:   writer ← csv.writer(csvfile) ▷ Opening and writing a CSV file
64:   for each flow in self.finishedFlows do
65:     flow ← flowType(flow) ▷ Getting the flow
66:     output ← flow.generateFlowFeatures() ▷ Getting the features
67:     writer.writerow(output) ▷ Writing features to CSV
68:   end for
69: end function
```

Fig. 3. Pseudocode for capturing stream data based on packet data from a pcap file.

In Fig. 4, flag features are also extracted to perform the calculations required for the research results. This data is written to a csv file after the data in the pcap file is completely read using the dumpFeatureToCSV function [29]. This process is performed each time a pcap file is read, and the data from each pcap file is saved to a separate csv file.

```
1: function init(generator, srcIP, dstIP, srcPort, dstPort, protocol, timeStamp, headBytes, payloadBytes, flags,
TCPWindow, payload)
2: id ← generator.nextId() ▷ Create a unique ID for the package
3: srcIP ← srcIP ▷ Setting the source IP address
4: dstIP ← dstIP ▷ Setting the destination IP address
5: srcPort ← srcPort ▷ Setting the source port
6: dstPort ← dstPort ▷ Setting the destination port
7: protocol ← protocol ▷ Protocol setup
8: timeStamp ← timeStamp ▷ Set timestamp
9: headBytes ← headBytes ▷ Setting the packet header length
10: payloadBytes ← payloadBytes ▷ Load length setting
.    .    .    .    .    .    .    .    .
.    .    .    .    .    .    .    .    .
80: return (flags >> 5) & 1 ▷ Check the URG flag
81: end function
82: function hasFlagECE()
83: return (flags >> 6) & 1 ▷ Check the ECE flag
84: end function
85: function hasFlagCWR()
86: return (flags >> 7) & 1 ▷ Check the CWR flag
87: end function
```

Fig. 4. Pseudocode for extracting features needed in a calculation.

B. Mathematical Representation of Markov Chain

Markov chains were first used by Andrey Markov in 1906 to show that the law of large numbers does not require that random variables be independent. Since then, Markov chains have been used in biology, finance, computer science, mathematics, and other fields [5, 21]. A Markov chain is a model that helps us visualize a sequence of random variables, states, and the states in which each of them can take on values from a given set. Markov chains can also be used to make predictions, predict future events, or understand a process that has occurred [22].

Consider a simple Markov chain with states and transition matrix $\|p_{ij}\|_1^n$. Given $r_1, \dots, r_n$, the terminal probabilities of the chain have the property of termination. The set of all states $\{n\}$ of the chain is decomposed into two disjoint subsets, that is, $\{n\} = \{n_1\} \cup \{n_2\}$. The transition from the initial set to the next one through $R_{1,2}$ and $R_{2,1}$ and vice versa from the final to the initial state is described by the Formula (1) below:

$$R_{1,2} = \sum_{\substack{i\in\{n_1\} \\ j\in\{n_2\}}} r_i p_{ij}\,,\ R_{2,1} = \sum_{\substack{i\in\{n_2\} \\ j\in\{n_1\}}} r_i p_{ij} \qquad (1)$$

Lemma. $R_{1,2} = R_{2,1}$ probability flows are balanced [36].

In [4], in particular, all the states in the first subset $\{n_1\}$ are numbered $1, \dots, k$, and all the positions of the Markov chain are numbered according to the chosen decomposition as $\{n_1\}, \{n_2\}$ (where k is the number of items in this subset). The items in the complement of the set $\{n\}$ belonging to the set $\{n_2\}$ are numbered $k+1, \dots, n$.

The final probabilities $r_1, \dots, r_n$ of the complete chain give rise to the relationship [Formula (2)]:

$$r_i = \sum_{j=1}^{n} r_j p_{ji}, all\ i = 1, \dots, n) \quad (2)$$

the i-number position is extracted in the following Formula (3):

$$-r_i(1 - p_{ii}) = \sum_{\substack{j=1 \\ j \neq i}}^{n} r_j p_{ji} = 0, i = 1, \dots, n) \quad (3)$$

The stochastic transition matrix $\|p_{ij}\|_1^n$ is Formula (4):

$$\sum_{j=1}^{n} p_{ij} = 1, i = 1, \dots, n \quad (4)$$

$(1 - p_{ii})$ expression is expanded as in Formula (5):

$$-r_i \sum_{\substack{j=1 \\ j \neq i}}^{n} p_{ji} + \sum_{\substack{j=1 \\ j \neq i}}^{n} r_j p_{ji} = 0, i = 1, \dots, n) \quad (5)$$

When passing to probability values, the first k of the above formula is added [Formula (6)]:

$$-\sum_{i=1}^{k} \sum_{\substack{j=1 \\ j \neq i}}^{k} r_i p_{ij} - \sum_{i=1}^{k} \sum_{j=k+1}^{n} r_j p_{ij} + \\ + \sum_{i=1}^{k} \sum_{\substack{j=1 \\ j \neq i}}^{k} r_i p_{ji} + \sum_{i=1}^{k} \sum_{j=k+1}^{n} r_j p_{ji} = 0 \quad (6)$$

The first term of the above Formula (6) is equal to the third term [Formula (7)].

$$\sum_{i=1}^{k} \sum_{\substack{j=1 \\ j \neq i}}^{k} r_i p_{ij} = \sum_{i=1}^{k} \sum_{j=1}^{k} r_j p_{ij} - \sum_{i=1}^{k} r_j p_{ii} = \\ = \sum_{j=1}^{k} \sum_{i=1}^{k} r_j p_{ji} - \sum_{i=1}^{k} r_j p_{ii} = \sum_{i=1}^{k} \sum_{\substack{j=1 \\ j \neq i}}^{k} r_j p_{ji} \quad (7)$$

(Here the sum indices are replaced equivalently by $i \leftrightarrow j$, and the order of the sums is changed.)

Using the above formulas, a generalized formula is obtained as in Formula (8):

$$-\sum_{i=1}^{k} \sum_{j=k+1}^{k} r_i p_{ij} + \sum_{i=1}^{k} \sum_{j=k+1}^{n} r_j p_{ij} = 0 \quad (8)$$

A sequence of random variables $X_0, X_1, X_2, \dots$ is called a Markov chain if the values of the variables are taken in the state space $\{1,2, \dots N\}$ and there exists a matrix $P = (p_{ij})$ of size $N \times N$ for any $n \geq 0$. $X_0, X_1, X_2, \dots$ denotes the set of states [Formula (9)].

$$P(X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \dots, X_0 = i_0) = \\ = P(X_{n+1} = j | X_n = i) = p_{ij} \quad (9)$$

P is called the transition matrix of the matrix chain, and P_{ij} is the probability of transition from i to j.

The transition probability $p_{ij} = P(X_{n+1} = j | X_n = i)$ is written in the form of a matrix of p_{ij} and is as follows [Formula (10)]:

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdot & \cdot & \cdot & p_{1n} \\ p_{21} & p_{22} & \cdot & \cdot & \cdot & p_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ p_{n1} & p_{n2} & \cdot & \cdot & \cdot & p_{nn} \end{bmatrix} \quad (10)$$

The sum of the transition probabilities $p_{11}, p_{12}, \dots, p_{1n}$ in each row must be equal to $p_{11} + p_{12} + \dots + p_{1n} = 1$ [Formula (11)].

$$\sum_{j=1}^{N} p_{ij} = \sum_{j=1}^{N} P(X_{n+1} = j | X_n = i) = 1 \quad (11)$$

where, the sum of all the possibilities in the row together is 100%.

So, if $X_0, X_1, X_2, \dots$ forms a Markov chain, then the transition probability of a matrix P of size $N \times N$ at step t is equal to $P(X_t = j | X_0 = i) = (P^t)_{ij}$.

If $X_0, X_1, X_2, \dots$ forms a Markov chain [11], then the probability distribution of X_0 of a matrix P of size $N \times N$ is equal to 1, and the probability distribution of a row vector π^T of size N is expressed as $\pi^T P^t$ [Formula (12)].

$$X_0 = \pi^T \Rightarrow X_t = \pi^T P^t \quad (12)$$

Stationary probability as in Formula (13):

$$\pi P = \pi \quad (13)$$

The condition $\sum \pi_i = 1$ must also be met, which indicates the probability of the process remaining in a given state for how long.

C. Formulation of Statistical Features of Network Flow with Unexpected Situation

Statistical features are used to detect unexpected situations in network traffic. The following statistical features are calculated for network traffic during the analysis:

- Mean as in Formula (14):

$$m_i = \frac{1}{n} \sum_{j=i}^{i+n} S_j \quad (14)$$

where, m_i is the mean value, n is the number of packets in the selected time interval, $S_i - j$ is the sample value of the traffic intensity at time, and the sum of values from $j = i$ to $i + n$ is obtained [1].

- Variance as in Formula (15):

$$D = \frac{1}{n-1} \sum_{j=i}^{i+n} \left(S_j - m_j\right)^2 \quad (15)$$

where, D is the variance, n is the length of the window under study (how many values were obtained), $S_i - j$ is the intensity at time (number of packets, traffic volume), m_j is the mean value [2].

- asymmetry coefficient as in Formula (16):

$$K_a = \frac{\frac{1}{n}\sum_{j=i}^{i+n}(S_j - m_j)^3}{D^3} \quad (16)$$

where, $S_i - j$ is the sample value at time (e.g., the number of packets), m_i is the mean value, D is the variance, and n is the number of values. The asymmetry coefficient represents the shift in the center of gravity of the statistical distribution [4].

- kurtosis coefficient as in Formula (17):

$$K_k = \frac{\frac{1}{n}\sum_{j=i}^{i+n}(S_j - m_j)^4}{D^4} - 3 \quad (17)$$

where, $S_i - j$ is the value at time, m_i is the mean, D is the variance, n is the number of values. The kurtosis or asymmetry coefficient measures the height or sharpness of the peak of a statistical distribution. In other words, it tells how far the values in the distribution are not far from or clustered around the mean [12].

- opposition coefficient as in Formula (18):

$$K_{oc} = \frac{1}{\sqrt{\eta}} \quad (18)$$

η is the excess parameter, defined as Formula (19):

$$\eta = K_k = \frac{\mu_4}{\sigma^4} \quad (19)$$

where, σ is the standard deviation (i.e. the square root of the variance), μ_4 is the fourth central moment (i.e. the average of the 4th order deviations of all values from the mean). The parameter K_{oc} varies from 0 (according to the Cauchy distribution) to 1 (discrete binomial distribution). For a normal distribution, $K_{oc} = \frac{1}{\sqrt{3}} \approx 0.577$.

Another parameter that characterizes the shape of the distribution is the entropy coefficient.

- entropy coefficient as in Formula (20):

$$K_{en} = \frac{\Delta_E}{\sigma} \quad (20)$$

where, Δ_E is the entropy value of the error, which means the level of uncertainty of the distribution, σ means the standard deviation, that is, the degree of deviation from the mean value [Formula (21)].

$$\Delta_E = \frac{1}{2} e^{H(\frac{X}{X_n})} \quad (21)$$

where, $H(\frac{X}{X_n})$ is the entropy (uncertainty) change introduced by measuring the random variable X, and e is a mathematical constant. The entropic error shows that the greater the uncertainty in the distribution, the greater the entropic error [Formula (22)].

$$H\left(\frac{X}{X_n}\right) = \int_{+\infty}^{-\infty} p(X) \ln\big(p(X)\big) dX \quad (22)$$

where, $p(X)$ is the probability density of the random variable [14]. Table I presents the entropic error distribution type.

TABLE I. ENTROPY COEFFICIENT DISTRIBUTION

Distribution type	K_{en} value	Note
Normal distribution	$K_{en} = \frac{\sqrt{2\pi e}}{2} \approx 2.066$	Maximum entropy
Uniform distribution	$K_{en} = \sqrt{3} \approx 1.73$	Average entropy
Cauchy distribution and discrete binomial distribution	$K_{en} = 0$	Very low entropy

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

Statistical analysis methods are used to record network traffic in a laboratory environment and identify unexpected events.

For verification, the recorded data was recorded at different times. The files in csv format were merged. Statistical analysis is performed based on this data.

A. Using Statistical Analysis Method

Fig. 5 shows the sample average value graph. In it, the sample average value is calculated based on Formula (14), $m_i = 1646.5714$, and according to the obtained result, the packet size in 9 flows was higher than the mean. This means that an unexpected situation occurred in the network flow.

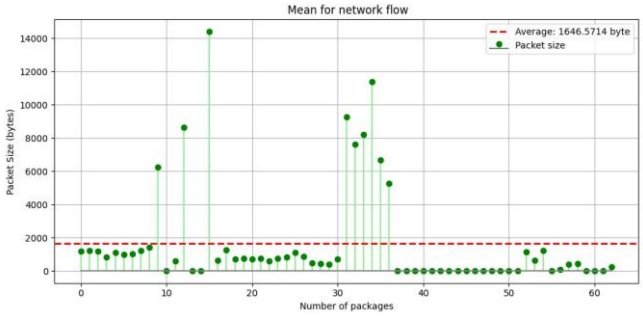

Fig. 5. Mean value graph of network flow.

In Fig. 6, the variance value was calculated based on Formula (15) and was equal to $D = 9474632.51^2$. In this statistical analysis, the square root of the variance of 9 network flows did not fall within the standard deviation range.

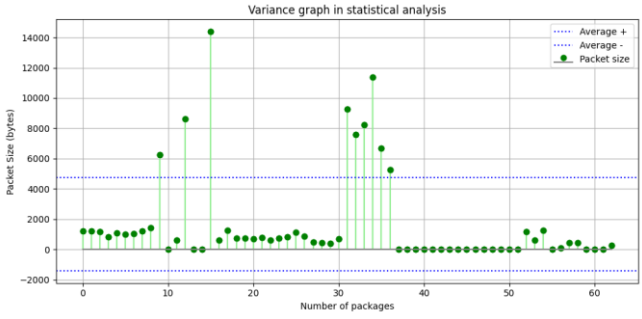

Fig. 6. Variance value graph of network flow.

The result of the asymmetry coefficient is presented in Fig. 7. The asymmetry coefficient is effective in determining how the data flow in the network is distributed, in which cases the packet size is small or large. This information is very important for

network optimization or security verification. The asymmetry coefficient helps to determine the presence of large packets in the network, which may indicate network congestion or problems. According to Formula (16), the asymmetry coefficient is equal to $K_a = 2.463632$. If the asymmetry coefficient is positive, this indicates a large number of large packets in the network and some uncertainties.

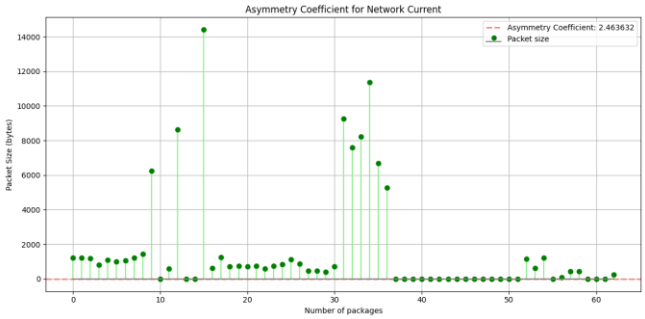

Fig. 7. Graph of the asymmetry coefficient of network flow.

The statistical distribution of the kurtosis coefficient can take very small or very large values. These results are useful in identifying unexpected situations in the network. In Fig. 8, the kurtosis coefficient is equal to $K_k = 8.352490$ according to Formula (17), which means that the value of the kurtosis coefficient is large. This indicates that there is an unexpected situation in the network.

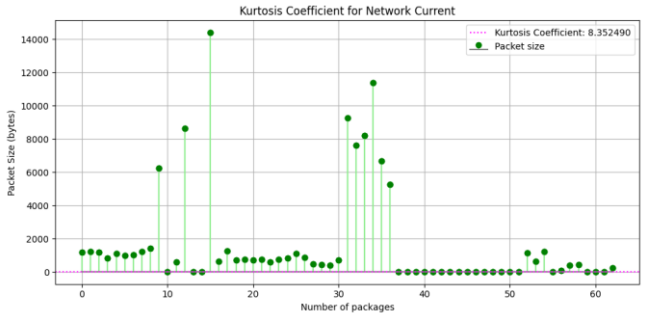

Fig. 8. Kurtosis coefficient graph of network flow.

The opposition coefficient is an indicator that provides deep statistical information about the symmetry and degree of deviation of the distribution in the network flow. The opposition coefficient is a very effective tool for identifying states in the network. The states that can be identified using K_{oc} are listed in Table II.

TABLE II. OPPOSITION COEFFICIENT CASES

Opposition coefficient	Distribution type	Note
$K_{oc} \approx 1.0$	Discrete, two-valued	Repeated and binary stream, 0 and 1500 byte packets, packets sent by test process or Botnet
$K_{oc} \approx 0.577$	Normal (Gaussian)	Stable network packet sizes are normally distributed
$K_{oc} < 0.4$	Heavy-tail	There are unexpected flows. For example, DoS or DDoS attacks, Botnets, etc.
$K_{oc} \approx 0$	Widely deviant distribution	Extremely dangerous, anomalous or exploitable condition

According to the result of the statistical analysis by recording the network current, the opposition coefficient is equal to $K_{oc} = 0.3516$. This indicates that there are unexpected situations in the network flow. In general, the combined use of statistical analysis results provides more accurate results. Fig. 9 presents the opposition coefficient result in graphical form.

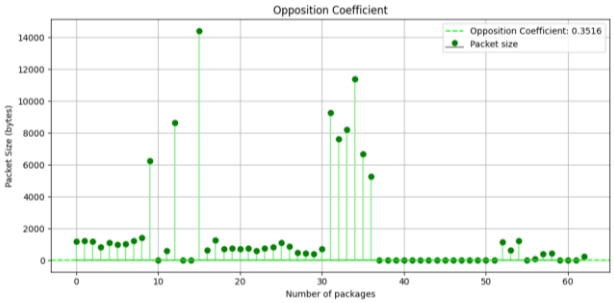

Fig. 9. Opposition coefficient in statistical analysis of network flow.

The entropy coefficient can be used to distinguish between normal and unexpected situations in a network. The entropy coefficient is an important statistical indicator for traffic analysis, attack detection, and security monitoring. As a result of the statistical analysis in Fig. 10, $K_{en} = 1.6231$. It indicates that there are different packets in the network flow, and there is an unexpected situation.

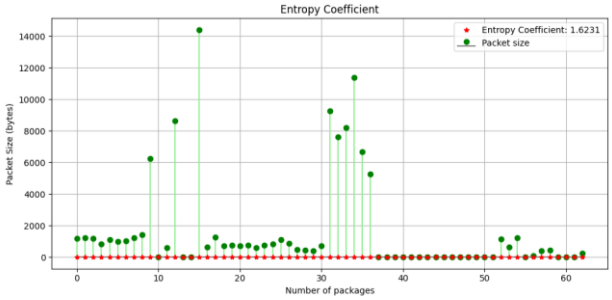

Fig. 10. Entropy coefficient in statistical analysis of network flow.

The analysis of the presented characteristics shows that the mean, variance, asymmetry coefficient, kurtosis coefficient, opposition coefficient, and entropy coefficient indicate the presence of unexpected situations in the network flow. Fig. 11 shows an overview of the research results.

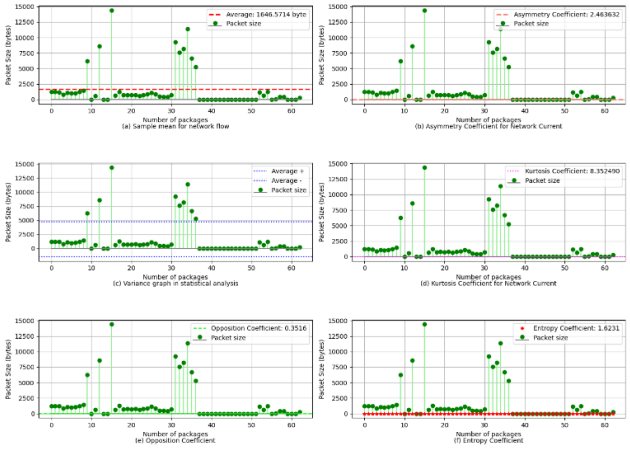

Fig. 11. General graph of statistical analysis results.

B. Markov Chain Construction Based on Statistical Analysis Results

Based on the results, statistical analysis methods are used to identify unexpected situations. The analysis of the obtained statistical characteristics shows that when an unexpected situation occurs in the network flow, the results of the mean and variance distinguish the unexpected situation, and the detection result may have an informative nature.

The values of the asymmetry coefficients and the kurtosis coefficient were large, and the values of the opposition coefficient and the entropy coefficient were low. According to the results of the study, these indicators are informative, since an unexpected situation occurs in the network flow. Table III shows the values of all statistical analysis results. The results obtained in the construction of the set of states of the Markov chain are the main value.

TABLE III. STATISTICAL ANALYSIS OF UNEXPECTED SITUATIONS IN NETWORK FLOW

№	Statistical analysis	The value of statistical analysis
1	Mean	$m_i = 1646.5714$
2	Variance	$D = 9474632.51^2$
3	Asymmetry coefficient	$K_a = 2.463632$
4	Kurtosis coefficient	$K_k = 8.352490$
5	Opposition coefficient	$K_{oc} = 0.3516$
6	Entropy coefficient	$K_{en} = 1.6231$

The Markov chain is evaluated based on the obtained statistical results. In the evaluation based on the Markov chain, based on the statistical analysis result, the first set of cases $X_0, X_1, X_2, \ldots$ is constructed, and the set of cases is presented in Table IV.

TABLE IV. SET OF STATES FOR A MARKOV CHAIN

№	Status name	Statistical analysis method
1	low load	variance, mean
2	average load	entropy coefficient, variance
3	high load	asymmetry coefficient, kurtosis coefficient

These three states were considered as states of the Markov chain, and based on the information obtained from the statistical analysis, the transition probability was calculated and a transition matrix was constructed according to Formula (1).

$$P = \begin{bmatrix} 0.4 & 0.4 & 0.2 \\ 0.2 & 0.4 & 0.4 \\ 0.1 & 0.2 & 0.7 \end{bmatrix}$$

The stationary probability was calculated using Formula (13). It was equal to $\pi = [0.1852 \quad 0.2963 \quad 0.5185]$.

Based on the transition probability matrix, Fig. 12 shows a graph of a Markov chain for network flow analysis. The graph shows the relationship between the three states.

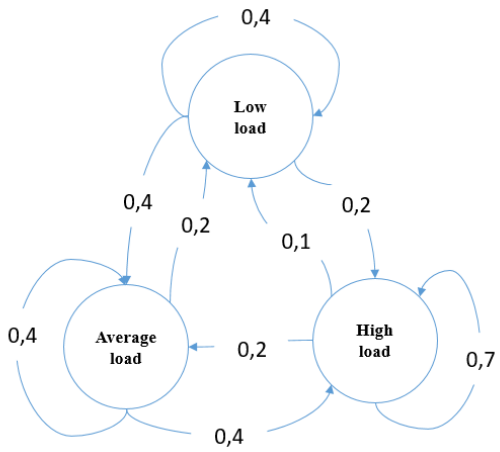

Fig. 12. Markov chain of network flow analysis.

VI. CONCLUSION

In solving the problem of detecting unexpected situations in network traffic, the use of statistical methods such as mean, variance, asymmetry coefficient, kurtosis coefficient, opposition coefficient and entropy coefficient supports the use of increasing efficiency and real-time operation. In detecting unexpected situations, packet features were extracted from the network traffic recorded in real time under laboratory conditions. Statistical analysis was performed using the extracted features. Markov chain is the most effective tool for analyzing situations and events occurring in network traffic and Markov chain was formed using the results of statistical analysis and Markov chain was constructed.

As a result, the low load stationary probability was 18.52%. This situation is rare in the system. This means that the network is in a low load state for a short time. The results of the statistical analysis correspond to the variance and mean, respectively. The average load stationary probability was 29.63%. This situation occurs moderately in the network flow. This means that there are changes, and the average load is frequent in the network. The high load stationary probability was 51.85%. The network flow time is more in this state. Therefore, the results of the statistical analysis are fully consistent with the results of the mean, variance, asymmetry coefficient and kurtosis coefficient. The Markov chain representing the probabilities of network states gave the same result as the statistical analysis.

In future work, using Markov chains in conjunction with statistical analysis methods will be effective. Markov chains are effective in showing the interaction between states when moving from one state to another. When used in conjunction with statistical analysis, the results of evaluating the object being analyzed will be more accurate.

REFERENCES

[1] Ramachandran K, Tsokos C. Mathematical Statistics with Applications in R, 2nd Edition. Elsevier, Inc.; 2015.

[2] Hu, C., & Liangping, H. (2022). How to use analysis of variance correctly—an analysis of variance for the univariate quantitative data collected from the orthogonal design. Sichuan Mental Health, 35(3), 201-206. doi:https://doi.org/10.11886/scjsws20220510004

[3] Alasmar, M., & Zakhleniuk, N. (2017). Network link dimensioning based on statistical analysis and modeling of real internet traffic. Ithaca: Retrieved from https://www.proquest.com/working-papers/network-link-dimensioning-based-on-statistical/docview/2076751109/se-2

[4] Chydzinski, A. (2025). Asymmetry between queues with correlated service and correlated arrivals. Symmetry, 17(1), 106. doi:https://doi.org/10.3390/sym17010106

[5] Kasse, M., Charrier, R., Berred, A., Bertelle, C., & Delpierre, C. (2024). An approach for anomaly detection in network communications using k-path analysis. Journal of Cybersecurity and Privacy, 4(3), 449. doi:https://doi.org/10.3390/jcp4030022

[6] Li, Y., & Wu, X. (2018). Packet forwarding strategies in multiagent systems: An evolutionary game approach. Wireless Communications & Mobile Computing (Online), 2018 doi:https://doi.org/10.1155/2018/4965343

[7] Zhou, K., & Nikaein, N. (2016). Random access with adaptive packet aggregation in LTE/LTE-A. EURASIP Journal on Wireless Communications and Networking, 2016, 1-15. doi:https://doi.org/10.1186/s13638-016-0531-6

[8] Sun, F., & Zhao, C. (2019). Research and modeling of photovoltaic array channel noise characteristics. Energies, 12(7), 1257. doi:https://doi.org/10.3390/en12071257

[9] Doncel, J. (2021). Age of information of a server with energy requirements. PeerJ Computer Science, doi:https://doi.org/10.7717/peerj-cs.354

[10] Ott, T. J., & Swanson, J. (2006). Asymptotic behavior of a generalized TCP congestion avoidance algorithm. Ithaca: Retrieved from https://www.proquest.com/working-papers/asymptotic-behavior-generalized-tcp-congestion/docview/2091799438/se-2

[11] Even, M. (2023). Stochastic gradient descent under markovian sampling schemes. Ithaca: Retrieved from https://www.proquest.com/working-papers/stochastic-gradient-descent-under-markovian/docview/2781016529/se-2

[12] Smiesko, J., Segec, P., & Kontsek, M. (2024). Machine recognition of DDoS attacks using statistical parameters. Mathematics, 12(1), 142. doi:https://doi.org/10.3390/math12010142

[13] Lu, F. (2024). Online shopping consumer perception analysis and future network security service technology using logistic regression model. PeerJ Computer Science, doi:https://doi.org/10.7717/peerj-cs.1777

[14] Kenyon, A., Deka, L., & Elizondo, D. (2024). Characterising payload entropy in packet Flows—Baseline entropy analysis for network anomaly detection. Future Internet, 16(12), 470. doi:https://doi.org/10.3390/fi16120470

[15] Gu, Y.; McCallum, A.; Towsley, D. Detecting anomalies in network traffic using maximum entropy estimation. In Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement, Berkeley, CA, USA, 19–21 October 2005; p. 32

[16] Doriguzzi-Corin, R., Luis Augusto, D. K., Mendozzi, L., Siracusa, D., & Savi, M. (2024). Introducing packet-level analysis in programmable data planes to advance network intrusion detection. Ithaca: Retrieved from https://www.proquest.com/working-papers/introducing-packet-level-analysis-programmable/docview/2836658859/se-2

[17] Abdulganiyu, O.H.; Ait Tchakoucht, T.; Saheed, Y.K. A systematic literature review for network intrusion detection system (IDS). Int. J. Inf. Secur. 2023, 22, 1125–1162. [CrossRef]

[18] Adrian-Tiberiu Costin, Zinca, D., & Dobrota, V. (2021). LAN traffic capture applications using the libtins library. Electronics, 10(24), 3084. doi:https://doi.org/10.3390/electronics10243084

[19] Taramit, H., José Jaime Camacho-Escoto, Gomez, J., Orozco-Barbosa, L., & Haqiq, A. (2022). Accurate analytical model and evaluation of wi-fi halow based IoT networks under a rayleigh-fading channel with capture. Mathematics, 10(6), 952. doi:https://doi.org/10.3390/math10060952

[20] Sembiring, I., Suharyadi, Iriani, A., Jenni Veronika, B. G., & Jusia, A. G. (2023). A novel approach to network forensic analysis: Combining packet capture data and social network analysis. International Journal of Advanced Computer Science and Applications, 14(3) doi:https://doi.org/10.14569/IJACSA.2023.0140353

[21] Kabulov, A.; Baizhumanov, A.; Saymanov, I. Synthesis of Optimal Correction Functions in the Class of Disjunctive Normal Forms. Mathematics 2024, 12, 2120. https://doi.org/10.3390/math12132120

[22] Kabulov, A.; Normatov, I.; Saymanov, I.; Baizhumanov, A. On the Completeness of Classes of Correcting Functions of Heuristic Algorithms, Azerbaijan Journal of Mathematics, 2025, vol 15, no. 2. (accepted)

[23] Kabulov, A., Baizhumanov, A., Berdimurodov, M. (2024). On the minimization of k-valued logic functions in the class of disjunctive normal forms. Journal of Mathematics, Mechanics and Computer Science, 121(1), 37–45. https://doi.org/10.26577/JMMCS202412114

[24] Kabulov, A.; Saymanov, I.; Babadjanov, A.; Babadzhanov, A. Algebraic Recognition Approach in IoT Ecosystem. Mathematics 2024, 12, 1086. https://doi.org/10.3390/math12071086

[25] Saymanov, I. . (2024). Logical automatic implementation of steganographic coding algorithms . Journal of Mathematics, Mechanics and Computer Science, 121(1), 122–131. https://doi.org/10.26577/JMMCS2024121112

[26] Islambek Saymanov, et al. Numerical Methods of Synthesis of a Correct Algorithm for Solving Recognition Problems. Advances in Artificial Intelligence and Machine Learning. 2025;5(1):202. https://dx.doi.org/10.54364/AAIML.2025.51202

[27] A. Kabulov, I. Normatov, E. Urunbaev and F. Muhammadiev, "Invariant Continuation of Discrete Multi-Valued Functions and Their Implementation," 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 2021, pp. 1-6, doi: 10.1109/IEMTRONICS52119.2021.9422486.

[28] A. Kabulov, I. Yarashov and A. Otakhonov, "Algorithmic Analysis of the System Based on the Functioning Table and Information Security," 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 2022, pp. 1-5, doi: 10.1109/IEMTRONICS55184.2022.9795746.

[29] Kabulov, A. V., Normatov, I. H. About problems of decoding and searching for the maximum upper zero of discrete monotone functions. Journal of Physics: Conference Series, 1260(10), 102006, 2019. doi:10.1088/1742-6596/1260/10/102006.

[30] A. Kabulov, I. Saymanov, I. Yarashov and A. Karimov, "Using Algorithmic Modeling to Control User Access Based on Functioning Table," 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 2022, pp. 1-5, doi: 10.1109/IEMTRONICS55184.2022.9795850.

[31] Kabulov, A. V., Normatov, I. H., Ashurov A.O. Computational methods of minimization of multiple functions. Journal of Physics: Conference Series, 1260(10), 10200, 2019. doi:10.1088/1742-6596/1260/10/102007.

[32] A. Kabulov, I. Normatov, A. Seytov and A. Kudaybergenov, "Optimal Management of Water Resources in Large Main Canals with Cascade Pumping Stations," 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Vancouver, BC, Canada, 2020, pp. 1-4, doi: 10.1109/IEMTRONICS51293.2020.9216402.

[33] E. Navruzov and A. Kabulov, "Detection and analysis types of DDoS attack," 2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 2022, pp. 1-7, doi: 10.1109/IEMTRONICS55184.2022.9795729.

[34] A. Kabulov, I. Saymanov, I. Yarashov and F. Muxammadiev, "Algorithmic method of security of the Internet of Things based on steganographic coding," 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 2021, pp. 1-5, doi: 10.1109/IEMTRONICS52119.2021.9422588.

[35] A. Babadzhanov, E. Urunbaev and I. Saymanov, "Problem of Synthesis of Minimal Forms of Logical Functions," 2022 International Conference on Information Science and Communications Technologies (ICISCT), Tashkent, Uzbekistan, 2022, pp. 1-5, doi: 10.1109/ICISCT55600.2022.10146903.

[36] Stefaniuk, V. L. (2011). Determinirovannye tsepi Markov. Information processes, 11(4), 423-427