

Research Article

A Random Traffic Assignment Model for Networks Based on Discrete Dynamic Bayesian Algorithms

Wei Zhou 

Information Engineering Department, Suzhou University, Suzhou 234000, China

Correspondence should be addressed to Wei Zhou; weizhou@ahszu.edu.cn

Received 6 December 2021; Revised 23 December 2021; Accepted 27 December 2021; Published 7 January 2022

Academic Editor: Gengxin Sun

Copyright © 2022 Wei Zhou. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, a stochastic traffic assignment model for networks is proposed for the study of discrete dynamic Bayesian algorithms. In this paper, we study a feasible method and theoretical system for implementing traffic engineering in networks based on Bayesian algorithm theory. We study the implementation of traffic assignment engineering in conjunction with the network stochastic model: first, we study the Bayesian algorithm theoretical model of control layer stripping in the network based on the discrete dynamic Bayesian algorithm theory and analyze the resource-sharing mechanism in different queuing rules; second, we study the extraction and evaluation theory of traffic assignment for the global view obtained by the control layer of the network and establish the Bayesian algorithm analysis model based on the traffic assignment; subsequently, the routing of bandwidth guarantee and delay guarantee in the network is studied based on Bayesian algorithm model and Bayesian algorithm network random traffic allocation theory. In this paper, a Bayesian algorithm estimation model based on Bayesian algorithm theory is constructed based on network random observed traffic assignment as input data. The model assumes that the roadway traffic distribution follows the network random principle, and based on this assumption, the likelihood function of the roadway online traffic under the network random condition is derived; the prior distribution of the roadway traffic is derived based on the maximum entropy principle; the posterior distribution of the roadway traffic is solved by combining the likelihood function and the prior distribution. The corresponding algorithm is designed for the model with roadway traffic as input, and the reliability of the algorithm is verified in the arithmetic example.

1. Introduction

With the development and popularity of internet technology, the internet has gradually affected every aspect of life, and various new application scenarios and network ecologies have emerged, promoting the rapid promotion and use of technologies such as big data and cloud computing [1]. The development of network scale gradually grows, new network technologies are diversified, and the traditional network architecture has difficulties in adapting to the deployment of new technologies [2]. In traditional networks, the routing addressing function is accomplished through the autonomous learning of routers, and the task of updating routing tables is dynamically maintained, further aggravating the burden of router forwarding devices. When the scale of the network expands, the network structure becomes

complicated, and the deep coupling of control protocols and routing and forwarding devices limits the horizontal scalability and openness of the traditional network, leading to increasingly complex network organization and management tasks. In the old network structure, to cope with various newly emerged application scenarios, additional corresponding functional services must be attached to the structure of forwarding devices such as routers, which are complex to configure and difficult to upgrade and optimize for the network [3]. At the same time, the traditional network through the router's autonomous learning to complete the route addressing function, dynamic maintenance of the routing table update task, further aggravates the burden of the router forwarding equipment. When the scale of the network expands, the network structure becomes complex, and the deep coupling of control protocols and routing and

forwarding devices limits the horizontal scalability and openness of the traditional network, leading to the increasingly complex task of network networking and management.

Bayesian modeling provides us with an elegant way to analyze large amounts of data and their hidden structure. Several research works have been performed using probabilistic graphical models for distributed Bayesian inference [4]. A distributed architecture for message passing based on network random traffic distribution is proposed, and this message propagation mechanism for network random traffic distribution is constructed by a minimal generation method [5]. By combining message propagation, a traffic distribution messaging mechanism on a network for the random network is proposed. However, this convergence is not guaranteed in graphs with rings and its computation is very difficult for complex models with non-Gaussian continuity variables [6]. For this reason, a nonparametric belief propagation algorithm is proposed by combining the ideas of Monte Carlo and particle filtering for modeling uncertainty. However, this sampling-based technique is computationally intensive and is not suitable for large-scale distributed datasets.

Regarding the application of Bayesian algorithm theory in SDN networks, end-to-end QoS analysis based on deterministic Bayesian algorithm is incorporated in the traffic distribution module of SDN switches by using real-time traffic information obtained in SDN. Various detection algorithms based on neural networks have good nonlinear fitting characteristics and can fully learn the complex nonlinear relationships that exist between data to detect attacks and normal traffic through data traffic characteristics. The performance of latency and queue length in OpenFlow switches is studied based on Bayesian algorithm theory, the analytical model of switches and controllers in controller separation architecture is analyzed, and the limiting performance of queue length in controllers is studied. In addition, the upper limit of queue length in the controller is studied by the Bayesian algorithm and then the scalability of the SDN network is studied. In mobile communication, SDN technology can define the same physical communication base station as several virtual base stations [7]. The end-to-end delay performance of data streams in virtual base station networks is investigated based on stochastic Bayesian algorithm theory. A new multitask detection system based on convolutional neural networks can achieve automatic anomaly learning using deep learning methods without the need to detect many traffic feature attributes in the system. The previous research results outline the Bayesian algorithm theoretical system; the network algorithm analysis model in SDN network architecture is studied, and the network algorithm conflict model under the control layer separation architecture and the analysis mechanism of network resource competition is proposed; the scheduling mechanism of three queuing rules, namely, first in, first out, random fair queuing, and hierarchical password bucket, is established in SDN based on the network algorithm.

2. Related Works

In traditional networks, network attack traffic detection is a classification problem, by which normal and attack traffics are distinguished. Because of the advantages of machine learning techniques in dealing with classification problems, the use of machine learning techniques in the field of intrusion detection to improve the detection effect is the focus of scholars' research, and there are numerous research results [8]. The research focuses on several aspects: first, using algorithms to select and optimize traffic data features to reduce the data dimensionality, eliminate redundant information, and improve the detection effect by obtaining the optimal feature subset; second, the selection of parameters when constructing models by machine learning affects the detection effect, introduces population intelligence optimization algorithms such as ant colony and particle swarm to search for the optimal parameters of the model, and builds a better classification model; meanwhile, many scholars explore the fusion of different machine learning algorithms to improve the detection effect [9]. Various detection algorithms based on neural networks have good nonlinear fitting characteristics, which can fully learn the complex nonlinear relationships existing between data and detect attacks and normal traffic through data traffic features [10]. The long- and short-term memory network structure is applied to recurrent neural networks to construct a three-layer recurrent neural network, the recurrent hidden layer achieves advanced abstraction of each feature attribute of the data by using complex architecture and nonlinear combinatorial transformation to spontaneously find correlations between network records for misuse detection, and the detection model obtained is tested using KDDCUP99 dataset training with good results [11]. A novel multitask detection system based on convolutional neural networks is proposed, which can achieve automatic anomaly learning using deep learning methods in a system where many traffic feature attributes do not need to be detected.

All paths in the random network allocation model are treated as determined utility values in the travel network, and different individuals perceive different utility values for the same travel path due to individual differences and the presence of random factors [12]. Several scholars have studied the selection probability of paths and the allocation of paths: first, for the problem of path selection encountered by passengers in travel, random utility theory is applied, and hypothesis is made that if the distribution of random factors in travel costs obeys the Grinnell distribution, then the network passenger flows are allocated by constructing the probit model and logit model. Dial was the first to apply the logit model to the problem of traffic flow allocation on urban roads, and later, as scholars continued to study the logit model, it was gradually used for passenger flow allocation in rail transportation [13]. The generalized cost model that includes the time of rides and transfers, the number of transfer factors is constructed, and the improved logit model is used for passenger flow allocation in transportation networks. Bayesian networks both qualitatively and quantitatively describe a real-world model; qualitatively, it

portrays the sum-dependency relationship between parent and child nodes through directed graphs, and quantitatively, it portrays the strength of this relationship through prior knowledge of the conditional probability distribution of variables. The shortcomings of the logit model and Dial's algorithm are considered, the fault tolerance coefficient index related to the length of the road section is considered, the effective path is defined, and then, the logit model is more efficiently solved using the improved Dial algorithm.

Bayesian algorithms are simple in structure and can handle various types of data and can be used to classify data with good results. Therefore, Bayesian methods have a wide range of applications in real life, such as text classification, spam classification, credit evaluation, and phishing website detection. The most widely used Bayesian classification model is the plain Bayesian approach, which assumes that attributes are independent of each other. With the increasing injection rate, the maintenance of the global topology information table is becoming more and more complex, and the routing algorithm based on the global topology information table cannot quickly determine the routing path of packets, which leads to delay, throughput rate performance degradation, and high-power consumption overhead. In contrast, the local topology-aware adaptive routing algorithm locally completes the reconfiguration information interaction and dynamically selects the available alternative optical nodes in the communication domain to transmit data based on the topology reconfiguration information. This assumption makes the logic of the classification algorithm simple, and its classification performance can even surpass that of classification algorithms such as SVM in many cases. When the dataset contains many attributes, there may be some redundant attributes among these attributes, which do not affect the final ground result, and even these redundant attributes add to the complexity of the classifier. Therefore, many scholars started to select the subsets of attributes appropriately, to select the subsets of attributes that have a strong connection with the categories, thus largely reducing the complexity of classification [14]. They used the forward selection method to select the optimal subset of attributes and used the plain Bayesian algorithm on the optimal subset of attributes. The Local topology-aware adaptive routing algorithm is used. The algorithm is based on the reservation mechanism of optical networks and selects optical nodes in the communication domain that are not reconfigured off for global communication when reconfiguration occurs, which effectively reduces communication delay and power consumption overhead and improves the communication performance of the network. The combination of forwarding selection and backward exclusion of the original attribute set not only improves the efficiency of the classification but also results in a more optimal subset of attributes. In response to the problem that Bayesian algorithms ignore the dependencies between attributes, many researchers have started to assign different weights to different attributes, and the weights can well reflect the influence that each attribute brings to the classification results. A weighted plain Bayesian approach, which assigns different weights to attributes based on the

gain rate, and attributes with higher gain rates are assigned higher weights. A decision tree and a plain Bayesian method are combined to estimate the weights of the attributes by constructing a decision tree. A differential evolution algorithm is used to calculate the weights. A weighting method based on eigenvalues is proposed, which can more accurately calculate the weights of each attribute.

3. Design and Construction of a Network Random Traffic Distribution Model Based on Discrete Dynamic Bayesian Algorithms

3.1. Discrete Dynamic Bayesian Algorithm System Construction. Bayesian networks are probabilistic graphical models that portray the dependencies between variables in the form of directed acyclic graphs. Nodes in a network graph represent random variables, and directed edges represent dependencies between variables, e.g., a directed edge from node A to node B can be expressed as "A causes B and B depends on A." Unconnected nodes indicate that they are conditionally independent of each other under certain conditions [15]. Each node contains a table of conditional probabilities calculated based on sample data. Thus, a Bayesian network both qualitatively and quantitatively describes a real-world model; qualitatively, it portrays the sum-dependency relationship between parent and child nodes through a directed graph, and quantitatively, it portrays the strength of this relationship through prior knowledge of the conditional probability distribution of the variables. In the responsive traffic matrix evaluation algorithm, the SDN controller extracts the core link as the monitoring link through the acquired network panorama view, then queries the traffic load information of the core link through the information carried by OpenFlow events and query commands and evaluates the traffic load on the aggregated link through the evaluation method, and then constructs a complete traffic matrix. A Bayesian network consists of two parts, namely, the directed graph and the conditional probability distribution of each node. Therefore, correspondingly constructing a Bayesian network also requires two steps of constructing the network combination and parameter learning. There are two methods for constructing Bayesian network structure, the first is manual construction, where experts in the field draw a network structure diagram by subjectively judging the dependencies between variables with their accumulated experience; the second is obtained through data, where a computer generates a network structure by discovering the laws of variable linkages from sample data according to a certain algorithm. When the number of samples is scarce, the first method based on subjective experience is suitable to construct the network structure, but when the amount of data is sufficient, the second method is commonly used, which does not require a lot of labor cost and is more objective. In addition, there are two ways to learn the parameters of Bayesian networks, which are the great likelihood estimation method and the Bayesian estimation method. When the sample size gets larger, the result of Bayesian estimation will converge more

and more with the great likelihood estimation. The Bayesian model construction process is shown in Figure 1.

Let the full set of variables in the original data be $X = \{X_1, X_2, X_3\}$. In the preparation phase, the input raw dataset is first divided into D_1 and D_2 . Then, a Bayesian network B with all variables in the set of all variables X_1 as nodes is constructed based on D . A node X_i is said to be a parent of X_j if there is a directed edge from node X_i to node X_j . Bayesian modeling provides us with an elegant way to analyze massive amounts of data and their hidden structure. Several research works have been performed using probabilistic graphical models for distributed Bayesian inference. A distributed architecture for message passing based on network random traffic assignment is proposed, and this message propagation mechanism for network random traffic assignment is constructed by a minimal generation method. All parent nodes of X are denoted as. Any node is conditionally independent of other nodes if its parent is known. Bayesian network is a probabilistic model that describes both continuous and discrete distributions [16]. Since discrete variables are more practical in credit data and their probability distributions are simpler to express, this study will focus on the discrete case. Therefore, assuming that all variables are discrete, the formula for the joint probability distribution corresponding to this Bayesian network is

$$P(X_e) = \sum \frac{P(X_e) - X_e}{p(e)}. \quad (1)$$

Traditional target tracking methods view a target only as a point source of measurement data. In this chapter, we assume that a target has a certain object shape and is described by a spatial model. We use a random matrix model as the spatial touch type in this study. It represents the moving target state by a random vector and the target profile by a symmetric positive definite random matrix, not ended, where d denotes the dimension of the target. The vector represents the position of the object and its motion properties, such as velocity, acceleration, and steering speed. The performance of latency and queue length in OpenFlow switches is studied based on Bayesian algorithm theory, and the analytical model of switches and controllers in controller separation architecture is analyzed to investigate the limiting performance of queue length in controllers. The symmetric positive definite matrix represents the shape of the object as an ellipse. The elliptical shape may seem to have limited expressiveness, but this model can be applied to many real-world scenarios, such as pedestrian tracking using LIDAR, and target tracking tasks in sensor networks using marine radar where the state of the extended target is collaboratively estimated by using the measurements collected by each node. The schematic diagram of the Bayesian algorithm is shown in Figure 2.

The Bayesian classification algorithm is a classification algorithm based on probability statistics, which can match or even surpass the performance of algorithms such as neural networks and decision trees in many contexts. The Bayesian algorithm is not only simple but also has a high

rate of correct classification when the size of the dataset is large. It can be precisely expressed in mathematical formulas. Bayesian algorithms have always been an important part of research in machine learning. Instead of simply classifying an instance to be detected into a certain class, Bayesian classifier classifies the instance to be detected into the class with maximum probability by calculating the probability of the instance to be detected belonging to each class and finally classifying the instance to be detected into the class with maximum probability. In the Bayesian classifier, all the attributes of the dataset are involved in the decision-making rather than a few attributes alone determining the result of the classification, and hence, the result obtained is relatively accurate. The Bayesian classification algorithm is shown in Figure 3. The most widely used Bayesian classification model is the plain Bayesian approach. The plain Bayesian classification approach assumes that the attributes are independent of each other, an assumption that makes the logic of the classification algorithm simple, and its classification performance can even outperform classification algorithms such as SVM in many cases.

The Bayesian algorithm is a probabilistic statistics-based approach to probabilistic inference, i.e., predicting the probability of a future event based on events that have already occurred. The main idea of Bayes' theorem is that if the probability of an event occurring is uncertain, then the only way to quantify it is the probability of the event occurring. The Bayesian classification algorithm is a classification algorithm based on probability statistics. On many occasions, the performance of the Bayesian classification algorithm can match or even surpass that of algorithms such as neural networks and decision trees. The Bayesian algorithm is not only simple but also has a high correct classification rate when the size of the dataset is large. If the probability of an event occurring is known, then the probability of a future event occurring can be mathematically calculated. Assuming a data sample space E , $P(A)$ denotes the probability of event A occurring, and $P(B)$ denotes the probability of event B occurring. $P(AB)$ denotes the probability of event A and event B simultaneously occurring. $P(B/A)$ denotes the probability of event B occurring if event A has already occurred, that is, ended.

$$P_B = \sqrt{\frac{P_{A-1} - P_{B+1}}{P_A}}, \quad (2)$$

From knowledge of mathematical probability theory, it follows that

$$P_A = \sqrt{\frac{P_{A+1} - P_{B-1}}{P_B}}. \quad (3)$$

Suppose that the events B_1, B_2, \dots, B_n are a complete set of events, i.e., all events are mutually exclusive and their sum set is the full set, and at the same time, the probability of each event is greater than 0. The probability of any event A occurring is given by the following equation:

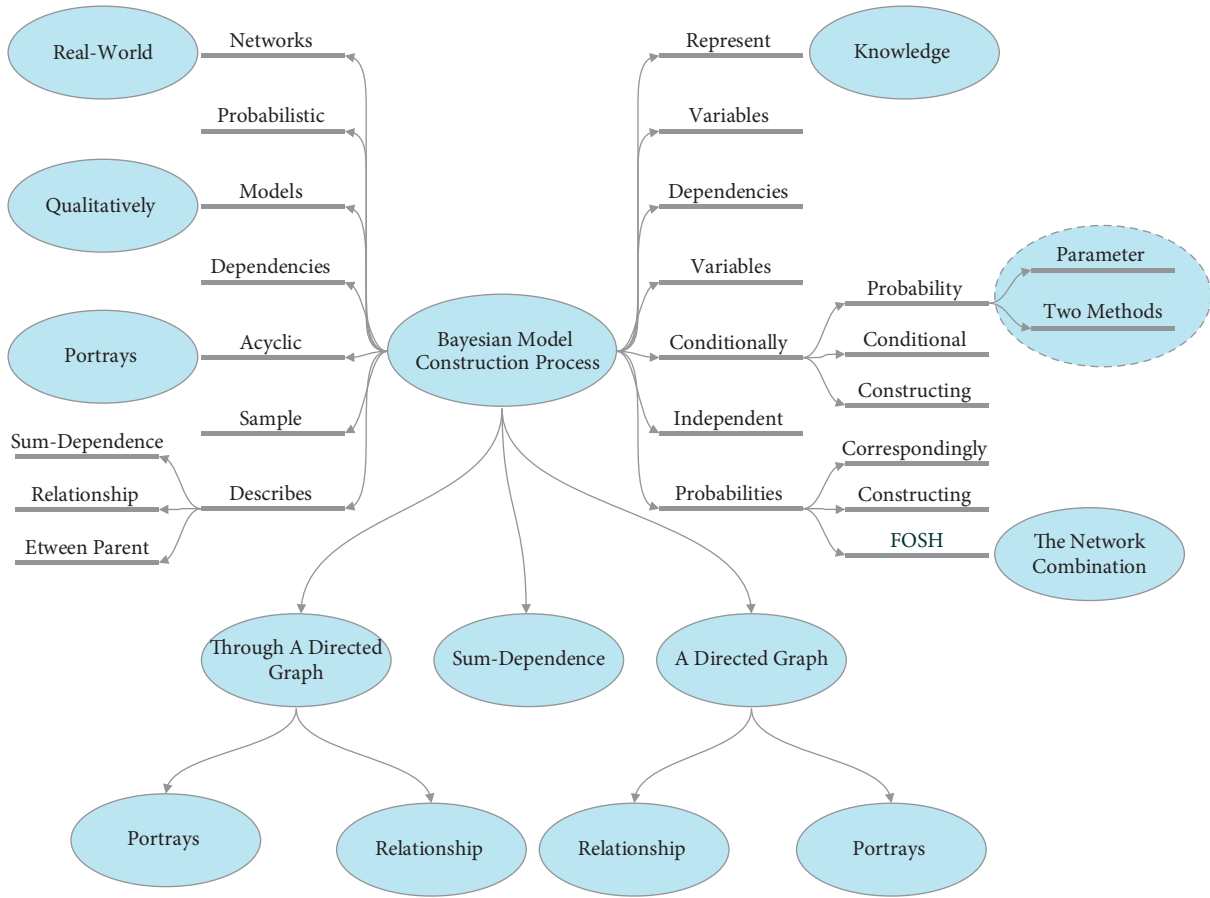


FIGURE 1: Bayesian model construction process.

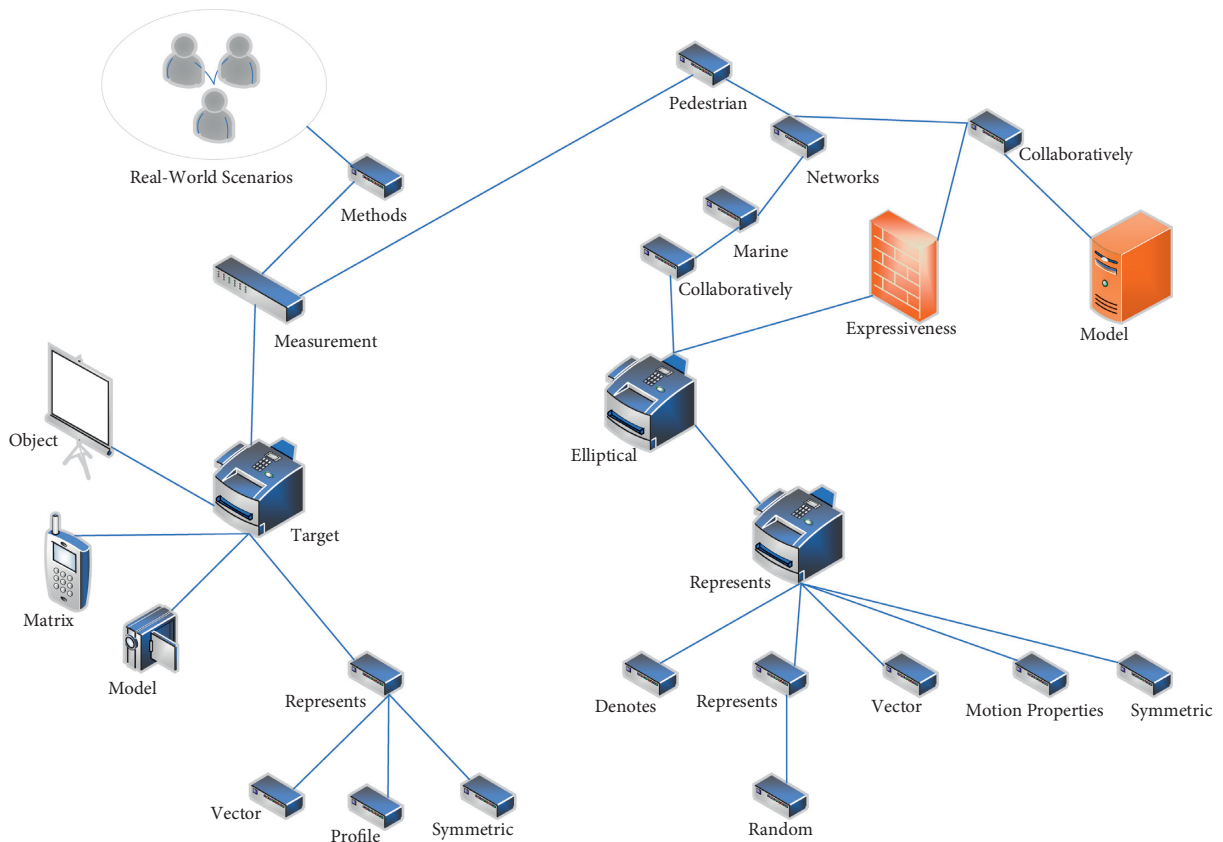


FIGURE 2: Schematic diagram of Bayesian algorithm.

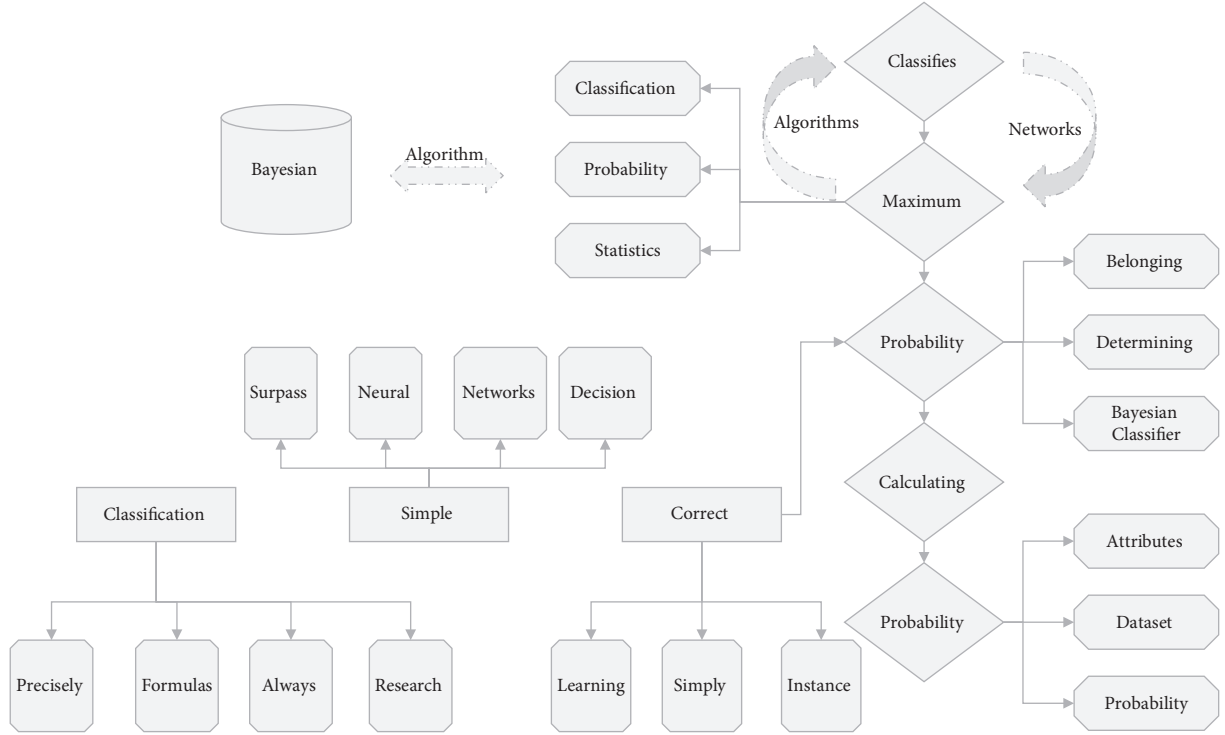


FIGURE 3: Bayesian classification algorithm.

$$P_{AB} = \sqrt{\frac{P_{A-1} - P_{B+1}}{P_A}} - \sqrt{\frac{P_{A+1} - P_{B-1}}{P_B}}. \quad (4)$$

Assuming that sample X is a sample data to be tested and B_i is a class in the sample space, the key to classification is to decide the most likely classification of sample X of data to be tested within the given data space. Bayes' theorem gives a way to calculate this likelihood. Bayes' theorem can be expressed in a mathematical formula, known as Bayes' formula. This is shown as follows:

$$P_{BX} = \sum \frac{P_{X-1} + P_{B+1}}{P_X - P_B}. \quad (5)$$

In Bayes' theorem, $P(X)$ denotes the probability of event X occurring and $P(B_i)$ denotes the prior probability of B_i . It is called the prior probability because its probability is not related to other events. $P(B_i)$ denotes the initial probability that B_i has, and usually, we use the ratio of the number of samples belonging to B_i in the sample space to the total number of samples to approximate this value, i.e., $P(B_i) = |B_i|/|D|$. Since $P(X_{ebio})$ is obtained from the probability of this, it is called the posterior probability of X . That is, what is the probability of the sample data X to be tested under the condition that the probability of B_i is known? Simply put, Bayes' theorem is based on prior probabilities and provides a method for computing posterior probabilities.

3.2. Network Random Traffic Assignment Model Design. The detection scheme utilizes the centralized control feature of the SDN network and the counter function of switch flow

table entries to extract corresponding features for attack behavior identification at different stages [17]. First, we monitor the resource utilization rate of the forwarding process v regularly switched of the network switch and trigger a "fine-grained" detection by monitoring its change to predetermine the switch with abnormal status. On top of that, based on the centralized control feature of SDN network architecture, the statistical analysis of Packet in messages sent by abnormal switches is performed to determine the attack behavior. Through the controller to obtain the switch flow table entry information to extract detection features, a machine learning algorithm-based classifier is input for attack traffic identification. The detection scheme uses the centralized control feature of the SDN network and the counter function of switch flow table entries to extract the corresponding features for attack behavior identification in different stages. The network intrusion attack detection mechanism mainly consists of the abnormal switch monitoring and locating phase, flow table request message feature detection, and flow table item feature extraction and identification phase, and the flow chart of detection mechanism triggered between each mechanism is shown in Figure 4.

3.2.1. Abnormal Switch Monitoring and Location. In this stage, the switch forwarding process detection algorithm warns the existence of the abnormal state of the switch, making the switch the basic unit of the monitoring network, monitoring the use of CPU resources of its forwarding process, responsible for coarse-grained monitoring of traffic access to the network via the switch forwarding device,

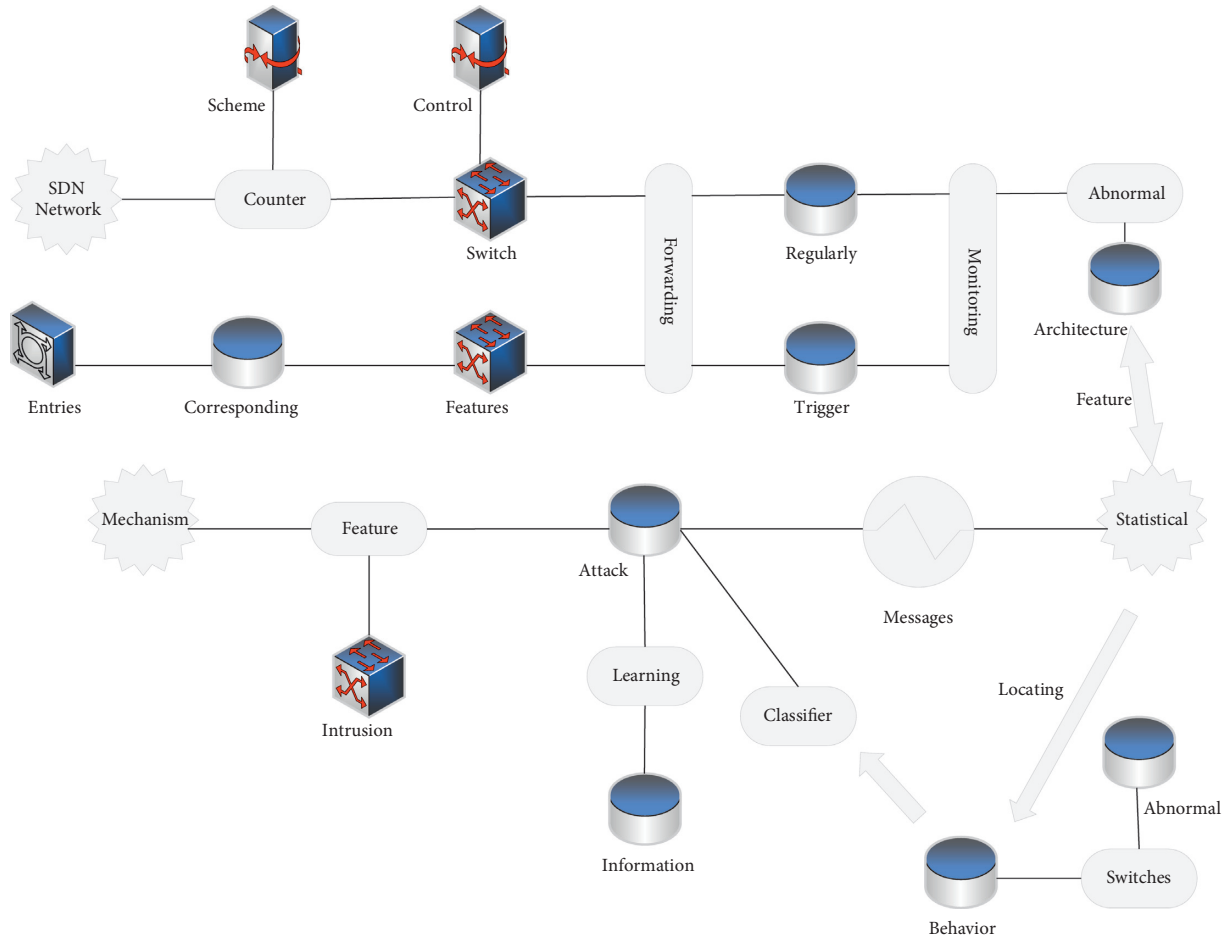


FIGURE 4: Flow chart of the detection mechanism.

initial detection and screening of all switches, and the next step of more fine-grained detection of abnormal switch network traffic. To achieve detection purposes, the access level of the forwarding process of the switch device is evaluated by the computational resource utilization rate within the window time, and when its change exceeds the threshold a certain number of times, with the help of an asynchronous message mechanism sent from the forwarding device to the controller, the SDN switch will encapsulate the alert message to the controller to trigger the next stage of detection. The forwarding process CPU monitoring module is mainly designed to narrow down the detection range of the intrusion detection system and alert the abnormal switch in time.

3.2.2. Flow Table Request Message Feature Detection. When the abnormal growth of traffic data in a switch device causes the resource usage of the switch device forwarding process to grow, and if it is caused by an external attack, the traffic is mostly new, causing the switch to generate a higher number of Packet in messages to the controller. The Packet in message-based attack detection module is responsible for monitoring the flow table entry mismatch traffic of the abnormal network forwarding device by collecting Packet in

messages from the abnormal network forwarding device for periodic aggregation and collation to detect the attack. By analyzing the number of Packets in packets and the cross-entropy value of the destination IP, it determines whether there is an attack on the network forwarding device.

3.2.3. Flow Table Item Feature Classification Detection Stage. The monitoring of the flow granularity level obtains the statistical information of the flow entry in the network switching device through the controller sending a control message and then extracts the feature to form the feature item. The data are input into the classifier to detect and identify the attack traffic. The attack detection model is constructed through the generalization capability of the Boost algorithm, while the uncertainty inference of the Bayesian idea is used to improve the selection stage of the algorithm model, which improves the detection accuracy of the classification model and ensures the accuracy of detecting attacks while shrinking the consumption of detection time. The controller middle layer module is responsible for interacting with the southbound application built in the controller and the upper layer business applications, the network access state monitoring module, and the flow table request message feature attack detection

module that is in the application layer and achieves their functions by calling the controller interface. The classifier module and the stream table entry message attack detection module are implemented as separate services that interact with the rest of the system modules in an interprocess communication manner.

$$\theta = \frac{\sqrt{a_k - m_k}}{\sum_k a_k - m_k}. \quad (6)$$

The SDN switch flow table entry has a counter module, and the OpenFlow plugin module inside the controller can periodically poll the SDN forwarding device for flow table statistic values through the flow statistic interface specified in the OpenFlow protocol and return the statistic values and store them in the database, and other applications can access this part of the database through the intermediate information [18]. The cycle statistics can be queried by accessing the controller's internal data store module, and the cycle detection application obtains the counter information of the flow table entries by calling the controller's externally open RPC interface, and the results of the switch port traffic statistics in the controller platform are shown in Table 1.

After receiving the issued detection command, the flow table item information in the forwarding device is periodically obtained through the controller for statistics, and the RPC interface open to the outside of the controller is called to return the value, where the data are stored in the open daylight inventory library, and the network forwarding device id, flow table id, and flow table item id to be queried need to be passed in as parameters. After obtaining the raw statistical information such as packet statistic value, packet bit statistic value, and duration of the flow, the information features are organized and input into the classifier module for a classification decision.

4. Analysis of Results

4.1. Analysis of Discrete Dynamic Bayesian Algorithmic Systems. Under reconfigurable optoelectronic hybrid on-chip networks, the global topology information table-based routing algorithm reaches saturation at an injection rate of 0.3, while the local topology-aware adaptive routing algorithm makes faster routing decisions due to its ability to adapt to dynamic changes in the topology, making the network saturate only at an injection rate of 0.4. When the injection rate is low, the amount of communication data is low and the difference in performance between the two algorithms is not significant. However, as the injection rate continues to increase, the maintenance of the global topology information table becomes increasingly complex, and the routing algorithm based on the global topology information table cannot quickly determine the routing path of packets, which leads to delay, throughput rate performance degradation, and high-power consumption overhead. The Bayesian algorithm is based on a discrete artificial bee colony algorithm for a structure learning problem. The Bayesian network structure learning problem is abstracted to the swarm search honey source problem in the artificial bee colony algorithm, and the directed acyclic graph is

transformed into the form of a matrix. In contrast, local topology-aware adaptive routing algorithms locally complete the reconfiguration information interaction and dynamically select the available alternative optical nodes in the communication domain to transmit data based on the topology reconstruction information [19]. When the latter is saturated, the local topology-aware adaptive routing algorithm reduces the network delay by 69.5%, improves the throughput rate by 33.3%, and reduces the network power consumption by 24.2% compared to the global topology information table-based routing algorithm. The results show that the local topology-aware adaptive routing algorithm improves the performance of network delay, throughput rate, and power consumption. The delay and throughput rate performance and the average power consumption overhead of the two routing algorithms are shown in Figure 5.

Mesh electrical interconnect network power consumption can be divided into two parts: static power consumption and dynamic power consumption, where dynamic power consumption mainly consists of link dynamic power consumption, routing dynamic power consumption, cache dynamic power consumption, and clock-driven power consumption, and static power consumption mainly consists of static link power consumption and static routing power consumption-NC, NRHOE-No C, and RHOE-No C power consumption, and in addition to the mesh-like electrical network part, there is also dynamic power consumption and static power consumption from the optical network. The dynamic power consumption mainly consists of microchip dynamic transmission power and photoelectric conversion interface dynamic cache power, and the static power consumption mainly consists of static laser source power, microring heating power, transmitter side power, and receiver side power. To analyze the effect of dynamic reconfiguration of optical nodes on the network power consumption, the experiments count the electric network dynamic power consumption, electric network static power consumption, optical network dynamic power consumption, and optical network static power consumption of four topologies, namely, mesh, HEO-NC, NRHOE-No C, and RHOE-No C, respectively. The injection rate is set as 0.2, and the power consumption composition of the four topologies is shown in Figure 6.

In describing the arrival and departure of network services to and from network nodes, a leaky bucket model has been proposed, where the speed of data arriving at the leaky bucket represents the arrival rate of data arriving at the network node, the capacity of the leaky bucket b represents the cache space of the network node, and the flow rate of the leaky bucket r represents the service capacity of the network node, where traffic accumulates in the leaky bucket when the speed of traffic arriving at the bucket is higher than the flow rate of the leaky bucket or when burst traffic arrives [20]. Overflow occurs when the accumulated traffic is higher than the capacity of the leaky bucket. In a network, a data flow is said to be nonconformant if there is an overflow occurring in this flow, and conversely, it is said to be conformant. Le bounded in his monograph provides an analytical model of

TABLE 1: The results of the switch port traffic statistics in the controller platform.

Controller platform to switch port traffic statistics results												
Node connector statistics for node id-openflow												
Node connector id	RX-Pkts	Tx-Pkts	Rx-bytes	Tx Bytes	Tx Bytes	Rx Drops	Rx Errs	Tx Errs	Rx OverRun Errs	Rx CRC Errs	Collisions	
openflow:1:1	156	156	14532	1	1	1	1	1	1	1	1	
openflow:1:2	94325	24323	14565734	0	0	0	0	0	0	0	0	
openflow:1:3	155	155	13245	1	1	1	1	1	1	1	1	
openflow:1:4	19767	143245	13452375	0	0	0	0	0	0	0	0	
openflow:1:5	158	158	13342	0	0	0	0	0	0	0	0	
openflow:1:6	10452	104532	176545537	0	0	0	0	0	0	0	0	
openflow:1:7	17545	164796	1334689	0	0	0	0	0	0	0	0	
openflow:1:8	159	159	156734	1	1	1	1	1	1	1	1	
openflow:1:9	166864	178543	1324353	0	0	0	0	0	0	0	0	

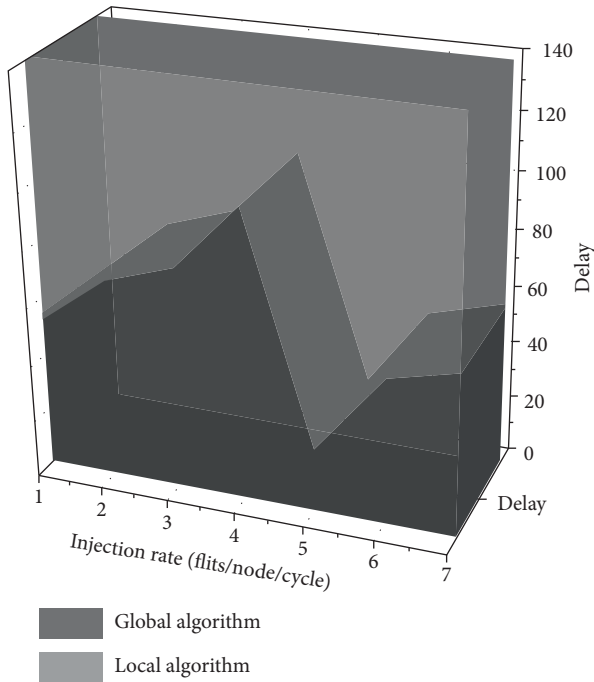


FIGURE 5: Delay throughput rate performance and average power overhead for two routing algorithms.

network evolution based on the least additive algebra for the leaky bucket model, in which the leading data flow of the leaky bucket model has γ_{rb} , as the arrival curve.

$$\partial_{rb} = \left\{ \begin{array}{l} \frac{rt - b}{R}, \quad R < 1, \\ \sum r - 1, \quad \frac{r}{t} - 1. \end{array} \right\} \quad (7)$$

In HTB, nodes share bandwidth using two models, namely, the password and the leaky bucket. In the HTB family, as shown in Figure 7, HTBs include root class (root class), interior class (interior class), and leaf class (leaf class). In HTB, classes are described by guaranteed rate AR (assured rate), maximum rate CR (ceil rate), priority P (priority), quantum Q (quantum), and level L (level). Data frames can only be transmitted to the output port through the class and only if there is a password

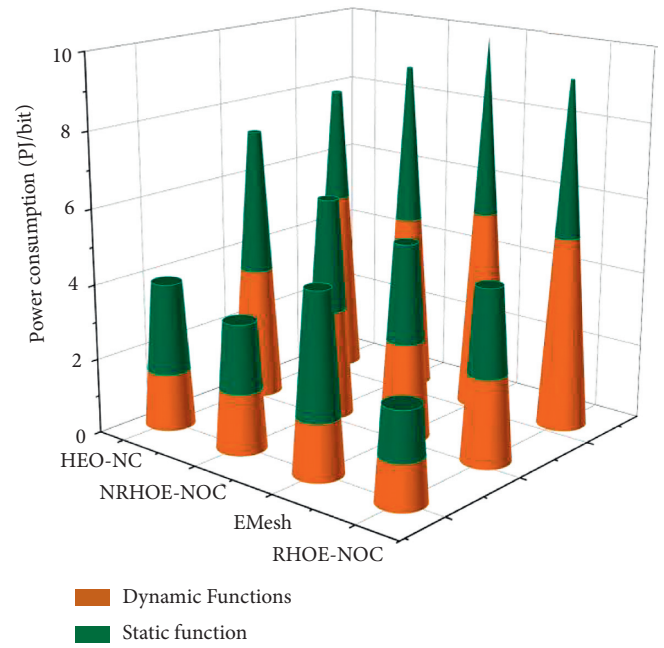


FIGURE 6: Power consumption composition diagram.

available for this leaf class. If the leaf class runs out of passphrases, then the leaf class can “borrow” a passphrase from its parent class, where the borrowed quotation marks indicate that the leaf node does not need to return the borrowed passphrase, even if the parent node has a passphrase shortage, and the leaf node will return the borrowed passphrase only when it has a spare passphrase after using the borrowed passphrase. The borrowing mechanism occurs under the following conditions: first, the data rate R of this leaf node satisfies $AR < R < CR$; second, the parent of this leaf node has a spare password resource; third, there is no higher priority business among the other children of its parent that needs to borrow the password, and in addition, if the parent is not the root node it can borrow the password from a higher-level node; and finally, several child nodes that can borrow the password are proportionally allocated the password lent by the parent node according to the Q value. The comparison graph of the leading data flow and its constraint curve in the HTB queuing rule is shown in Figure 7.

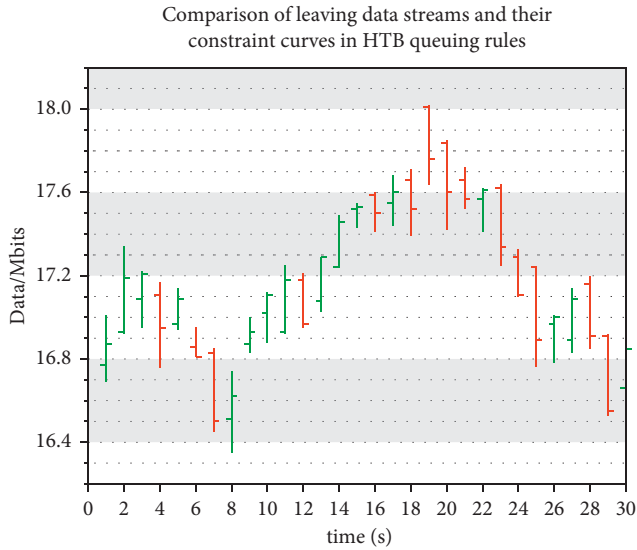


FIGURE 7: Comparison of leaving data streams and their constraint curves in HTB queuing rules.

SFQ queuing rules attempt to provide a fair allocation mechanism for any number of data streams in the network. The implementation principle is to use a hash function to assign all data streams to each FIFO queue, and the service node will use polling (round-robin) for these FIFO queues to serve each FIFO queue in turn, using weighted polling. In the SFQ mechanism, the greatest possibility of unfairness comes from the choice of hash functions, and some network applications can take favorable parameter settings to obtain more resources according to the nature of the hash functions, so the hash functions in the SFQ mechanism are also periodically changed. The service profile of each data stream in the SFQ queuing model is when the data streams are served with equal probability and are served for equal time, and all three data streams receive service with weight ϕ .

$$\beta_i = \frac{\partial - 1}{3\partial_{t+1}} \beta(t + 1). \quad (8)$$

4.2. Network Random Traffic Distribution Test Results. A Bayesian algorithm is based on a discrete artificial bee colony algorithm for a structure learning problem. First, the Bayesian network structure learning problem is abstracted to the swarm search honey source problem in the artificial bee colony algorithm, and the directed acyclic graph is transformed into the form of a matrix. Then, for the problem that the search space is a discrete domain, the hybrid differential evolutionary discrete artificial bee colony algorithm MABC-BNL is proposed to incorporate crossover and variation operators into the artificial bee search strategy for new food sources, while a new selective variation operator is designed, thus balancing the exploratory and exploitative nature of the algorithm. Furthermore, a structure graph correction scheme is designed to correct for emerging food sources that may appear in the search process using a depth-first search algorithm for “illegal” structures with loops. By establishing

the Markov chain model of food sources in MABC-BNL and combining the transfer probability of states, the algorithm is theoretically analyzed to converge to the optimum according to probability 1. Finally, the parameters in MABC-BNL are selected through simulation experiments and compared with several structure learning algorithms to verify the effectiveness of the algorithm. A hybrid constraint-based structure learning method will be given for the shortcomings of the evolutionary computation-based Bayesian network structure learning method. See Figure 8 for a comparison of convergence under different training sets.

The traffic matrix reflects the traffic load of the whole network and can provide a decision basis for studies such as network planning, traffic engineering, and routing optimization. In recent years, researchers have proposed some traffic matrix information extraction and evaluation models based on SDN graphs but providing a traffic matrix extraction scheme with low monitoring loss and high monitoring and evaluation accuracy in SDN switches via the OpenFlow protocol is still a research challenge, and there is still less research on how to effectively apply the traffic matrix to traffic engineering and routing optimization. In this chapter, a responsive traffic matrix extraction and evaluation algorithm is implemented in SDN networks based on the event information of the OpenFlow protocol. This convergence is not guaranteed in graphs with loops and is computationally difficult for complex models with non-Gaussian continuous variables. For this reason, a non-parametric belief propagation algorithm is proposed by combining the ideas of Monte Carlo and particle filtering for modeling uncertainty. In the responsive traffic matrix evaluation algorithm, the SDN controller extracts the core links as monitoring links through the acquired panoramic view of the network, then queries the traffic load information of the core links through the information carried by OpenFlow events and query commands and evaluates the traffic load on the aggregated links through the evaluation method, and then constructs a complete traffic matrix. On top of the algorithm for traffic matrix extraction and evaluation, this chapter proposes an algorithm for the traffic history matrix based on the distribution characteristics of the traffic time-space obtained in the Abilene backbone network, which records the characteristics of the time-space distribution of the entire network traffic based on the responsive algorithm in this chapter. As for how the traffic matrix can be effectively used in terms of routing optimization, this chapter proposes the theory of network evolution martyring, which uses the information of the traffic matrix for the construction of the network evolution service matrix. In this part of the study, based on the traffic matrices obtained by the controller, the pre-existing traffic in the network is used as the conflicting flows for the new arrival traffic, and the theory of network-acting target and conflicting flows is used to build the service curve and the remaining service curve for the output port of each switch node, and then, the service matrix and the remaining service matrix are built by combining the traffic matrices. Based on the routing matrix provided by the routing algorithm, end-to-end service curves can be computed for each routed link and used to

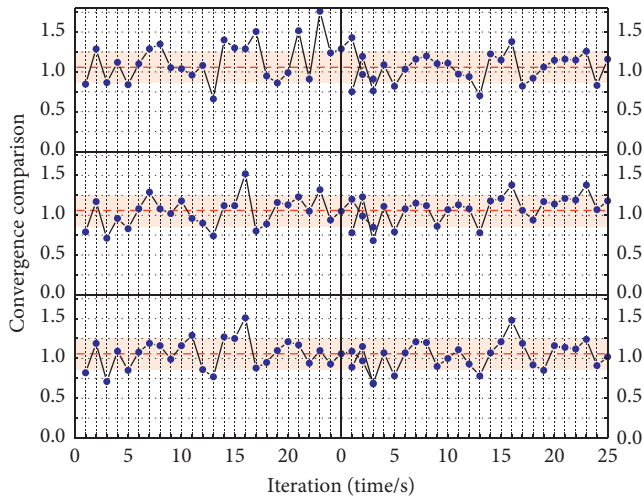


FIGURE 8: Comparison of convergence under different training sets.

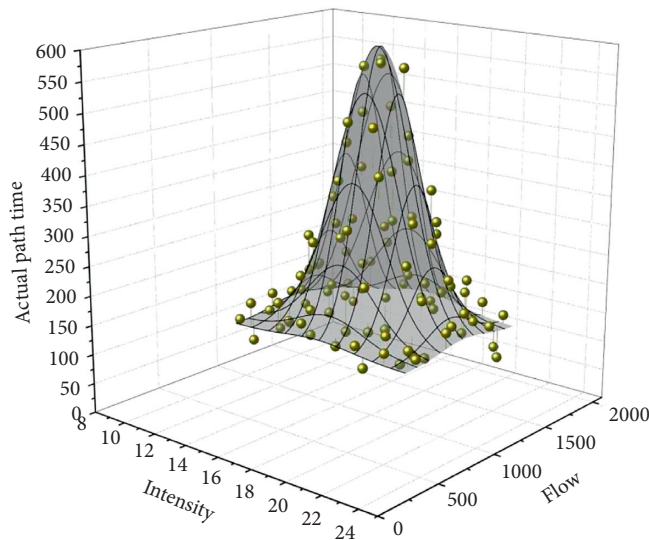


FIGURE 9: Graphical representation of actual path travel time.

analyze the upper bound on the delay of the data flow. This part of the work develops analytical theorems for matrix-based delay upper bounds and traffic backlogs, resulting in a theoretical system for traffic matrices to guide end-to-end QoS guarantees. A graphical representation of the actual path travel time is shown in Figure 9.

The traditional routing algorithm faces problems such as high delay and high-power consumption in reconfigurable topology, and the interaction mechanism of reconfiguration information is analyzed, based on which a local topology-aware adaptive routing algorithm is proposed. The algorithm is based on the reservation mechanism of optical networks and selects optical nodes in the communication domain that are not reconfigured off for global communication when reconfiguration occurs, which effectively reduces the communication delay and power consumption overhead and improves the communication performance of the network. In the AODV routing protocol in WSNs, the AODV routing

protocol initiates route discovery through a flooding process before route establishment and then establishes routes based on the received reply information that is suitable for dynamic network environments. This chapter is based on the principle of AODV protocol to establish routes that are more stable for high-performance links by evaluating the energy and load of the discovered routes between route discovery and route establishment. The work based on SDN-WISE uses Python to implement an SDN controller, verifies the working principle of software-defined WSNs, proposes a resource-intensive routing protocol, and tests metrics such as hop count and transmit/receive duty cycle between source and destination under different topologies in conjunction with a simulation platform. The results show that the programmability of network management brought by SDN for WSNs networks is of significant positive significance for resource-constrained wireless sensor nodes.

5. Conclusions

With the rapid development of internet technology, software-defined networking technology has been applied in cloud data centers, wide area networks, and other scenarios due to its advantages of high networking flexibility and scalability, but the network still has security problems to be solved. In this study, we study the problem of traffic distribution detection and identification in networks, draw on the advantages of security detection strategies in traditional networks, combine the characteristics of network randomness, make improvements to the shortcomings, and propose a multilevel phased hybrid detection method. Compared with the traditional rule-based inference methods, Bayesian algorithms can deal with complex, fuzzy, and uncertain problems very well. As a probabilistic inference model combining graph theory and probability theory, Bayesian algorithms are widely used in fault diagnosis, risk assessment, and situational assessment. The study of Bayesian algorithm mainly includes structure learning, parameter learning, and inference, where structure learning is also the basis for subsequent parameter learning and inference, and building an effective and accurate network structure can make the analysis clearer and the inference more accurate. The traditional method of constructing Bayesian algorithms based on expert knowledge is to consult experts and then use voting and scoring to determine the directed edges in the network. This method of consulting experts can be time-consuming and resource-intensive, and when the Bayesian algorithm is more complex, the expert knowledge-based construction method will undoubtedly pose great difficulties. Compared to expert knowledge, the method of constructing Bayesian algorithms from historical training data is more efficient and the learned network traffic assignment structure is more relevant to practical applications. However, since the possible structure of the Bayesian algorithm exponentially increases with the number of network nodes, which leads to the problem of low accuracy or high complexity of the methods learned through data, good structure learning algorithms are needed to improve the accuracy of the structure and the efficiency of learning.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank the College of Modern Industry (szxy2021cyxy04); 2019 Teaching and Research Project of the Suzhou University (szxy2019jyxm12); the Key Disciplines of Computer Science and Technology (2019xjzdxk1); and 2020 Teaching and Research Project of the Anhui Provincial Department of Education (2020jyxm2220).

References

- [1] M. Jiang and J. Lu, "Maritime accident risk estimation for sea lanes based on a dynamic Bayesian network," *Maritime Policy & Management*, vol. 47, no. 5, pp. 649–664, 2020.
- [2] X. Zhang, D. Rey, S. T. Waller, and N. Chen, "Range-constrained traffic assignment with multi-modal recharge for electric vehicles," *Networks and Spatial Economics*, vol. 19, no. 2, pp. 633–668, 2019.
- [3] Q. Xiao, Y. Zhao, and W. Huan, "Multi-sensor data fusion for sign language recognition based on dynamic Bayesian network and convolutional neural network," *Multimedia Tools and Applications*, vol. 78, no. 11, pp. 15335–15352, 2019.
- [4] J. Zhang, Y. Zheng, J. Sun, and Q. Dekang, "Flow prediction in spatio-temporal networks based on multitask deep learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 3, pp. 468–478, 2019.
- [5] Y. Xing, C. Lv, and D. Cao, "Personalized vehicle trajectory prediction based on joint time-series modeling for connected vehicles," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, pp. 1341–1352, 2019.
- [6] A. R. Pitombeira-Neto, C. F. G. Loureiro, and L. E. Carvalho, "A dynamic hierarchical Bayesian model for the estimation of day-to-day origin-destination flows in transportation networks," *Networks and Spatial Economics*, vol. 20, no. 2, pp. 499–527, 2020.
- [7] M. Salvato, J. Buchner, T. Budavári et al., "Finding counterparts for all-sky X-ray surveys with NWAY: a Bayesian algorithm for cross-matching multiple catalogues," *Monthly Notices of the Royal Astronomical Society*, vol. 473, no. 4, pp. 4937–4955, 2018.
- [8] H. M. Torun, M. Swaminathan, A. Kavungal Davis, and M. L. F. Bellaredj, "A global Bayesian optimization algorithm and its application to integrated system design," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 26, no. 4, pp. 792–802, 2018.
- [9] D. Calvetti, A. Pascarella, F. Pitolli, E. Somersalo, and B. Vantaggi, "Brain activity mapping from MEG data via a hierarchical Bayesian algorithm with automatic depth weighting," *Brain Topography*, vol. 32, no. 3, pp. 363–393, 2019.
- [10] A. Durmus and E. Moulines, "High-dimensional Bayesian inference via the unadjusted Langevin algorithm," *Bernoulli*, vol. 25, no. 4, pp. 2854–2882, 2019.
- [11] S. Oh, R. Seshadri, C. L. Azevedo, and M. E. Ben-Akiva, "Demand calibration of multimodal microscopic traffic simulation using weighted discrete SPSA," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2673, no. 5, pp. 503–514, 2019.
- [12] M. Rahbar, M. Hickman, M. Mesbah, and A. Tavassoli, "Temporal validation of a multimodal transit assignment model," *Case Studies on Transport Policy*, vol. 8, no. 2, pp. 535–541, 2020.
- [13] D. Sha, K. Ozbay, and Y. Ding, "Applying Bayesian optimization for calibration of transportation simulation models," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2674, no. 10, pp. 215–228, 2020.
- [14] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: a survey," *The International Journal of Robotics Research*, vol. 39, no. 8, pp. 895–935, 2020.
- [15] S. Scheubner, A. T. Thorgeirsson, M. Vaillant, and F. Gauterin, "A stochastic range estimation algorithm for electric vehicles using traffic phase classification," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6414–6428, 2019.
- [16] Z. Gu, S. T. Waller, and M. Saberi, "Surrogate-based toll optimization in a large-scale heterogeneously congested network," *Computer-Aided Civil and Infrastructure Engineering*, vol. 34, no. 8, pp. 638–653, 2019.
- [17] M. Scutari, "Bayesian network models for incomplete and dynamic data," *Statistica Neerlandica*, vol. 74, no. 3, pp. 397–419, 2020.
- [18] J. Du, C. Jiang, J. Wang, Y. Ren, and M. Debbah, "Machine learning for 6G wireless networks: carrying forward enhanced bandwidth, massive access, and ultrareliable/low-latency service," *IEEE Vehicular Technology Magazine*, vol. 15, no. 4, pp. 122–134, 2020.
- [19] M. Husák, J. Komárková, E. Bou-Harb, and P. Čeleda, "Survey of attack projection, prediction, and forecasting in cyber security," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 640–660, 2018.
- [20] A. Alsirhani, S. Sampalli, and P. Bodorik, "DDoS detection system: using a set of classification algorithms controlled by fuzzy logic system in Apache spark," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 936–949, 2019.