IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 1, 2020, accepted August 14, 2020, date of publication August 18, 2020, date of current version August 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3017673

Parameter Optimization Model of Heuristic Algorithms for Controller Placement Problem in Large-Scale SDN

YI LI, SHAOPENG GUAN, CONGHUI ZHANG, AND WENWEN SUN
School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai 264005, China

Corresponding author: Shaopeng Guan (konexgsp@gmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61976125, and in part by the Graduate Science and Technology Innovation Project of Shandong Technology and Business University under Grant 2019YC020.

ABSTRACT The Controller Placement Problem (CPP) is a key technical challenge in a large-scale Software Defined Network (SDN). Low-complexity heuristic algorithm is widely used for solving the CPP. However, parameter settings of the heuristic algorithm greatly affect the result of the CPP. Therefore, we establish a Parameter Optimization Model (POM) for the heuristic algorithm applied to the CPP. The heuristic algorithm can effectively solve the CPP by using the optimized parameters obtained in POM. To verify the effectiveness of the POM, we first establish a synthetical-delay controller placement model to reduce the delay between the controllers and the switches and the delay between the controllers. Further, we select the bat algorithm, the firefly algorithm, and the varna-based optimization respectively to solve the model, and use the particle swarm optimization method to optimize the parameters of the three algorithms. Experimental results on real topologies show that compared with original algorithms and other similar algorithms, the algorithms with optimized parameters perform better.

INDEX TERMS Controller placement problem, delay, heuristic algorithm, parameter optimization, particle swarm optimization, software defined network.

I. INTRODUCTION

With the rapid development of the communications technology, the number of Internet users is increasing day by day. In terms of network scalability and reliability, a wide variety of services make higher requirements that traditional networks cannot meet. Thus, Software Defined Network (SDN) came into being [1]. Separation of the coupled control and forwarding functions is the distinctive feature of the SDN. The centralized control layer improves the reliability and the robustness of the network. For a small-scale SDN, placing one controller is sufficient. However, with the expansion of the network scale, a single controller can increase the delay between some switches and the controller. Furthermore, once the controller fails, the entire network would be paralyzed. In short, multiple controllers need to be placed to manage a large-scale SDN [2].

To place multiple controllers in the SDN, the first thing is to determine locations of the controllers and mapping relationship between the controllers and the switches. Heller *et al.* named this problem Controller Placement Problem (CPP). They pointed out that the CPP is an NP-hard problem of which an optimal solution cannot be achieved within a reasonable period of time [3]. For the original CPP model, they took into account only the delay between the controllers and the switches and used the brute force method to traverse all possible conditions. To improve the robustness and the reliability of the SDN, studies of the CPP after [3] contain more indicators, such as the delay between the controllers, the load of the controllers, and the load of the switches [4], [5]. With the increase of network indicators, it is unrealistic to use brute force to obtain the optimal solution. For example, in [3], only the delay between the controllers and the switches was considered, and the author spent several weeks using the brute force method to analyze all possible controller placements schemes. In [6], the author used the enumeration method to solve a variant of the CPP, which includes the delay between

The associate editor coordinating the review of this manuscript and approving it for publication was Noor Zaman.

the controllers, the load of the controllers, and the load of the switches. Due to the time-consuming problem, this research was only limited to the small and medium SDN.

The heuristic algorithm belongs to the stochastic algorithm. The design mechanism of the heuristic algorithm does not depend excessively on the problem. A near-optimal solution can be obtained within a reasonable period of time. Therefore, the heuristic algorithm is suitable for solving complex engineering problems [7], [8]. The inspiration of designing the heuristic algorithm usually comes from nature. For example, Particle swarm optimization (PSO) is inspired by ethnic behavior of birds [9]; Bat algorithm (BA) is inspired by foraging behavior of bats [10]; Firefly algorithm (FA) is inspired by courtship behavior of fireflies [11]; Teaching-learning-based optimization (TLBO) is inspired by teaching behavior of human beings [12]; and varna-based optimization (VBO) is inspired by social structure and production behavior of human beings [13]. As a result, the heuristic algorithm is more and more used for solving the CPP due to its low-time complexity and its suitability for solving high-dimensional problems [17]–[25].

In a large number of existing studies, the heuristic algorithms are simply improved or directly applied to solve the CPP. However, if the key parameters of the heuristic algorithm are improperly set, the solution results will be badly affected. In light of this, we proposed a parameter optimization model (POM) of the heuristic algorithm for solving the CPP. The main contributions are as follows:

- To meet different controller placement requirements, we took the weighted sum of the delay between the controllers and the switches and the delay between the controllers as the synthetical delay. Then, with the minimum synthetical delay as the goal, we put forward the synthetical-delay controller placement model (SDCPM).
- We established the POM that obtains optimal parameters of the controller placement algorithm to better solve the SDCPM.
- To verify the effectiveness of the POM, we designed the SDCPM solution algorithm based on the FA, the BA, and the VBO, respectively, and then used the PSO to optimize the parameters of the FA, the BA, and the VBO.

The rest parts of this article are organized as follows: Section II presents the related work, followed by the model and method in section III. Section IV evaluates the model through experiments. Finally, conclusions are presented in section V.

II. RELATED WORK

Since the CPP was formally proposed, different network indicators have been introduced into the CPP to meet different SDN controller placement requirements. Compared with traditional network, the most distinctive feature of the SDN is that data packets are forwarded by the SDN in exchanging the flow table. The flow table in each switch is delivered by the controller. Frequent communications are performed between the controllers and the switches. Therefore, the delay between the controllers and the switches is the most important CPP indicator [14]–[37]. After being placed with multiple controllers, the SDN is divided into multiple subdomains, and each subdomain is managed by one controller. The information of links and nodes in the SDN needs to be unified among multiple controllers. To ensure consistency between the controllers in the SDN, communication between the controllers is also required. Therefore, the delay between the controllers is also a key CPP indicator [14]–[29]. Besides, placement cost [21], network fault tolerance [28], path failure probability [30] and the like, are common CPP indicators.

From the perspective of the solving method used in CPP, a tendency was shown from using the brute force method for the optimal solution to using a stochastic algorithm for the near-optimal solution. In [7], the author used the brute force method to analyze multiple CPP indicators. Being affected by time consumption, the experiment was limited to the small and medium SDN. In [40], the author designed a controller placement algorithm based on an optimal algorithm [41], but high-time complexity exists. With the expansion of the network scale and the increase of various indicators, the brute force method and the optimization algorithm are unable to produce the optimal solution in a reasonable period of time. Further then, scholars turned their focus on using the stochastic algorithm with low-time complexity to solve the CPP, for example, clustering-based controller placement algorithms, which includes density-based clustering [16], basic clustering [26], spectral clustering [31], affinity propagation [33]. These clustering-based controller placement algorithms first divide the entire SDN into multiple subnets and then determine the location of each controller to minimize the delay between the controllers and the switches. However, more controller placement algorithms were designed based on the heuristic algorithm, for example, the controller placement algorithm based on the physical law heuristic algorithm: simulated annealing algorithm [15]; the controller placement algorithm based on the swarm intelligent behavior heuristic algorithm, which includes the PSO [18], FA [25], moth flame optimization [27], and adaptive bacterial foraging optimization [30]; the controller placement algorithm based on the human behavior heuristic algorithm, which includes TLBO [20] and VBO [29]; and the controller placement algorithm based on the evolutionary behavior heuristic algorithm, which includes non-dominated sorting genetic algorithm [17], non-dominated sorting genetic algorithm II [22], and genetic algorithm [32]. The controller placement algorithm based on the heuristic algorithm usually simulates the coordinates of the controllers which need to be placed in the network into a multi-dimensional object in the heuristic algorithm, uses the location update strategy, sets the objective function, and iteratively solves the CPP. Due to the excellent performance in resolving the multi-dimensional problem and the multi-objective optimization problem, the heuristic algorithm is very suitable for solving the CPP in the large-scale SDN.

The existing controller placement algorithm based on the heuristic algorithm usually applies a heuristic algorithm to CPP directly. However, the result of the heuristic algorithm for solving the CPP is greatly affected by the initial parameters. If the parameters are not properly set, the solution will not meet the placement requirements. In this article, we considered the two most important CPP indicators: the delay between the controllers and the switches and the delay between the controllers. Taking the weighted sum of the two delays as the synthetical delay, we established the SDCPM, which can flexibly adjust the proportion of the two delays to meet different placement needs. Further, to obtain the optimal parameters of the controller placement algorithm, we established POM, which aims to use the parameter-optimized controller placement algorithm to obtain a controller placement scheme with a lower synthetical delay.

III. MODEL AND METHOD

The process, which applies the controller placement algorithm optimized by POM to solve the SDCPM, can be divided into three phases. In phase one, the heuristic algorithm applied to the SDCPM and its key parameters to be optimized are determined. In phase two, a random SDN topology as an instance is selected and the solving algorithm for the POM is determined. Then, the near-optimal parameters of the controller placement algorithm are output. In phase three, the parameters-optimized controller placement algorithm is used to solve the SDCPM in other topology instances. The specific process is shown in Figure 1:

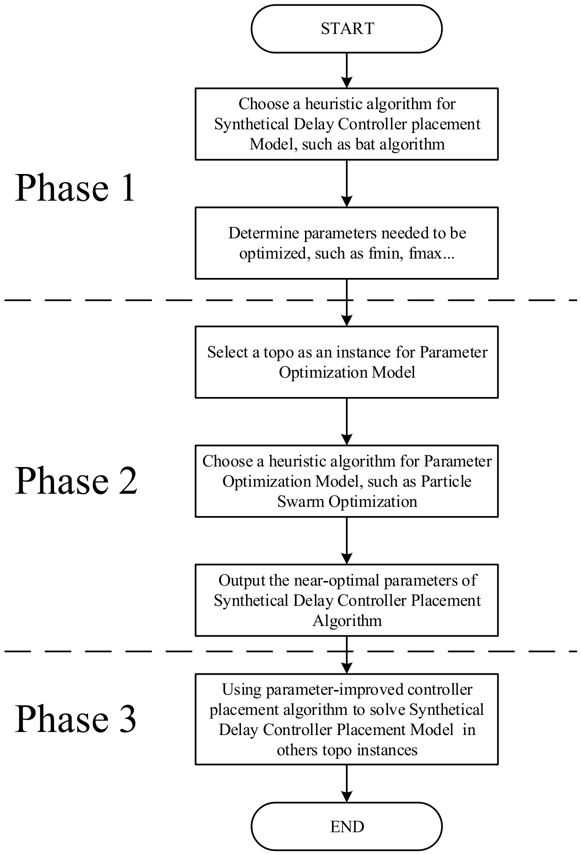

START
Choose a heuristic algorithm for Synthetical Delay Controller placement Model, such as bat algorithm
Phase 1
Determine parameters needed to be optimized, such as fmin, fmax...
Select a topo as an instance for Parameter Optimization Model
Phase 2
Choose a heuristic algorithm for Parameter Optimization Model, such as Particle Swarm Optimization
Output the near-optimal parameters of Synthetical Delay Controller Placement Algorithm
Phase 3
Using parameter-improved controller placement algorithm to solve Synthetical Delay Controller Placement Model in others topo instances
END

FIGURE 1. Applying the controller placement algorithm optimized by parameter optimization model to solve the synthetical-delay controller placement model.

A. SYNTHETICAL-DELAY CONTROLLER PLACEMENT MODEL

We modeled the large-scale SDN topology as an undirected graph G (S, L), where S is the set of switch nodes in the topology (the quantity is N), and L is the set of links between the switches, which are shown in formula (1) and (2):

$$S = \{s_i | i = 1, 2, \ldots, N\}. \tag{1}$$

$$L = \left\{link\left(s_i, s_j\right) | s_i, s_j \subset S\right\}. \tag{2}$$

where, s_i and s_j are switch nodes, $link\left(s_i, s_j\right)$ is the link between the s_i and the s_j.

C is the set of controller nodes in the topology (the quantity is K), and L_C is the set of links between the controllers, which are shown in (3) and (4):

$$C = \{c_i | i = 1, 2, \ldots, K\}. \tag{3}$$

$$L_C = \left\{link\left(c_i, c_j\right) | c_i, c_j \subset C\right\}. \tag{4}$$

where, c_i and c_j are controller nodes, $link\left(c_i, c_j\right)$ is the link between the c_i and the c_j. In-band placement mode is adopted, that is, the controller node is placed on the inherent switch node, and the link between the switches is used, that is $C \subset S, L_C \subset L$.

The assigned controller of the switch is the controller with the closest link distance from the controller set, which is shown in (5):

$$B(s_i) = \min(link_{dst}(s_i, c_j)), \quad j = 1, 2, \ldots, K. \tag{5}$$

where, $B(s_i)$ represents the assigned controller of the switch s_i, min represents a minimum function, and $link_{dst}(s_i, c_j)$ represents the link distance of switch s_i and controller c_j.

The mapping relationship between the controllers and the switches is shown in (6):

$$Map = \{B\left(s_i\right) | s_i \subset S\}. \tag{6}$$

where, Map represents the mapping relationship, $B(s_i)$ represents the assigned controller of the switch s_i.

The set of the switches contained in the controller is shown in (7):

$$c_i^{cont} = \left\{s_j | B\left(s_j\right) = c_i\, j = 1, 2, \ldots, N\right\}. \tag{7}$$

where, c_i^{cont} represents the set of the switches contained in c_i.

The average delay between the controllers and the average delay between the controllers and the switches are shown in (8) and (9):

$$D_{CC}^{mean} = \left(\sum_{i=1}^{K} \sum_{j=1}^{K} link_{dst}(c_i, c_j) \right) / (K-1) \times K \ \times H. \quad (8)$$

where, D_{CC}^{mean} represents the average transmission delay between the controllers, $link_{dst}(c_i, c_j)$ represents the link distance between the c_i and the c_j, K represents the number of controllers, and H represents the data transmission speed of the link.

$$D_{CS}^{mean} = \left(\sum_{i=1}^{N} link_{dst}\left(s_i, B\left(s_i\right)\right) \right) / N \times H \ . \quad (9)$$

where, D_{CS}^{mean} represents the average transmission delay between the controllers and the switches, $link_{dst}\left(s_i, B\left(s_i\right)\right)$ represents the link distance between the switches and its assigned controllers, N represents the number of switches, and H represents the data transmission speed of the link.

The two aforementioned delays reflect different placement requirements. Then, we propose a synthetical delay, which is shown in (10):

$$\text{DELAY} = \lambda_1 D_{CC}^{mean} + \lambda_2 D_{CS}^{mean}. \quad (10)$$

where, DELAY represents the synthetical delay, λ_1 and λ_2 are two weight coefficients of two kinds of delay and $\lambda_1 + \lambda_2 = 1$. Two coefficients may be adjusted according to different controller placement requirements.

The SDCPM is used to find the controller placement scheme with a minimum value of DELAY, which is shown in (11):

$$Min \ (\text{DELAY}), \quad s.t. \, C \subset S, L_C \subset L. \quad (11)$$

where, Min is the minimum function.

B. PARAMETER OPTIMIZATION MODEL

The POM of the controller placement algorithm is shown in (12):

$$\begin{aligned} &Min \ (result), result = \sum_{1}^{n} CPA\,(a, b, c)/n \\ &s.t. \ a \subset \left[a_{low}, a_{up}\right], b \subset \left[-\infty, b_{up}\right], c \subset \left[c_{low}, \infty\right]. \end{aligned} \quad (12)$$

where, CPA represents a controller placement algorithm. a, b and c respectively represent three variable initial parameters of the CPA. The number of variable initial parameters of the CPA may not be three. For ease of description, we assume that there are three variable initial parameters. Each parameter needs to comply with the constraints of the CPA. The parameters with both upper and lower limits, such as a, are taken a value between the upper limit a_{up} and the lower limit a_{low}. The parameters with only upper limit, such as b, need to be within a range less than the upper limit b_{up}. The parameters with only lower limit, such as c, need to be within a range higher than the lower limit c_{low}. $CPA\,(a, b, c)$ represents the return value of the CPA solving the SDCPM, that is, the synthetical delay of the controller placement scheme. Further then the $CPA\,(a, b, c)$ is repeated n times, and the average value is used as the objective function (e.g. $result$) of the POM. POM needs to minimize the value of the result under the constraints of the parameters.

C. SYNTHETICAL-DELAY CONTROLLER PLACEMENT ALGORITHM

To verify the effectiveness of the POM, we designed the synthetical-delay controller placement algorithms based on the FA, the BA, and the VBO respectively.

1) BA-BASED SYNTHETICAL-DELAY CONTROLLER PLACEMENT ALGORITHM

BA is a heuristic algorithm that simulates the use of sonar by bats to detect the prey and avoid obstacles in nature [10]. BA-based synthetical-delay controller placement algorithm (BACPA) is shown in Algorithm 1.

Algorithm 1 BACPA

```
Input: iteration times: T, number of bats: n, topology information: G, initial parameters variable of the BA: $f_{min}$, $f_{max}$, $\beta$, $\gamma$, $R_0$, $A_0$
Output: $x_{optimal}$ and corresponding $Map$
1:  Randomly generate vector $x$, vector $v$
2:  According to the coordinates of switches, discrete $x$
3:  According to $R_0$ and $A_0$, generate initial n-dimensional vector $R$ and $A$
4:  According to (6), calculate the $Map$
5:  According to (10), calculate the $DELAY$
6:  for t in (0, T) do
7:      Calculate $A_{mean}$, $x_{optimal}$, and $DELAY_{optimal}$
8:      $v = v_{old} + (x_{old} - x_{optimal})^*$ rand($f_{min}$, $f_{max}$)
9:      $x = x_{old} + v$
10:     Calculate the $Map$ and the $DELAY$
11:     if rand(0, 1) > R: then
12:         $x_{new} = x_{optimal} +$ rand(−1, 1)*$A_{mean}$
13:     end if
14:     $x_{new} = x_{old} +$ rand(−1, 1)*$A_{mean}$
15:     Calculate $Map$ and $DELAY$ for $x_{new}$
16:     if rand(0, 1) < $A$ and $DELAY < DELAY_{optimal}$ then
17:         $x = x_{new}$
18:         $A = \beta$*$A$
19:         $R = R$*(1 − exp(−$\gamma$*t))
20:     end if
21: end for
22: Output $x_{optimal}$ and corresponding $Map$
```

Firstly, initialize the key variables of the algorithm and randomly generate n sets of controller coordinate vector x, and velocity vector v (lines 1 to 5). Next, find the optimal solution iteratively in the entire solution space (lines 6 to 21). Finally, output the optimal solution of the controller coordinates $x_{optimal}$, and the corresponding Map (line 22).

rand (a, b) in Algorithm 1 means that random numbers between a and b are taken. For the description of the key formulas in the BA, refer to [10].

2) FA-BASED SYNTHETICAL-DELAY CONTROLLER PLACEMENT ALGORITHM

FA is a heuristic algorithm that simulates the courtship behavior of fireflies in nature [11]. FA-based synthetical-delay controller placement algorithm (FACPA) is shown in Algorithm 2.

Algorithm 2 FACPA

```
Input: iteration times: T, number of bats: n, topology information: G, initial parameters variable of the FA: $\alpha$, $\beta$, $\gamma$
Output: $x_{optimal}$ and corresponding Map
 1: Randomly generate vector $x$
 2: According to the coordinates of switches, discrete $x$
 3: According to (6), calculate the Map
 4: According to (10), calculate the DELAY
 5: for t in (0, T) do
 6:     for i in (0, n) do
 7:         for j in (0, n) do
 8:             $B[j] = \beta * DELAY[j] * \exp(norm(x[i], x[j]))$
 9:         end for
10:         Calculate the index Des[i] of maximum B
11:         $x[i] = x_{old}[i] + \beta * \exp(-\gamma * norm(x[i], x[j])) * (x[Des[i]] - x_{old}[i]) - \alpha * Gauss(0, 1)$
12:     end for
13:     Calculate the Map and the DELAY
14:     Update $x_{optimal}$ and corresponding Map
15: end for
16: Output $x_{optimal}$ and corresponding Map
```

Firstly, initialize the key variables of the algorithm (lines 1 to 4). Next, find the optimal solution iteratively in the entire solution space (lines 5 to 15). Finally, output the optimal solution of the controller coordinates $x_{optimal}$, and the corresponding *Map* (line 16). Gauss (a, b) in Algorithm 2 means that random numbers on a normal distribution with an expectation of 0 and a standard deviation of 1, are taken. For the description of the key formulas in the FA, refer to [11].

3) VBO-BASED SYNTHETICAL-DELAY CONTROLLER PLACEMENT ALGORITHM

VBO is a heuristic algorithm that simulates the social structure and production behavior of human beings [13]. VBO-based synthetical-delay controller placement algorithm (VBOCPA) is shown in Algorithm 3.

Firstly, initialize the key variables of the algorithm (lines 1 to 5). Next, find the optimal solution iteratively in the entire solution space (lines 6 to 24). Finally, output the optimal solution of the controller coordinates $x_{optimal}$, and the corresponding *Map* (line 25). R (0, 1) in Algorithm 3 represents multidimensional vector composed of random numbers from 0 to 1. For the description of the VBO, refer to [13].

Algorithm 3 VBOCPA

```
Input: iteration times: T, number of fireflies: n, topology information: G, initial parameters variable of the VBO: $\alpha, c1, c2$
Output: $x_{optimal}$ and corresponding Map
 1: Randomly generate vector $x$
 2: According to the coordinates of switches, discrete $x$
 3: According to (6), calculate the Map
 4: According to (10), calculate the DELAY
 5: $na = \alpha * n$
 6: for t in (0, T) do
 7:     $x_{optimal}$ and $x_{worst}$
 8:     for i in (0, na) do
 9:         $x[i] = x_{old}[i] + R(0, 1) * (x_{optimal} - x_{worst})$
10:     end for
11:     for i in (na,n) do
12:         Generate a random integer $r \neq i$ from 0 to n
13:         $x_{peer} = x[r]$
14:         if $DELAY[i] < DELAY[r]$ then
15:             $x[i] = x_{old}[i] + c1 * R(0,1) * (x_{optimal} - x_{peer})$
16:         else if $DELAY[i] > DELAY[r]$ then
17:             $x[i] = x_{old}[i] + c2 * R(0,1) * (x_{peer} - x_{old}[i])$
18:         else
19:             $x[i] = 2 * rand(0,1) * x_{old}[i]$
20:         end if
21:     end for
22:     Calculate the Map and the DELAY
23:     If current solution performs better, then accept
24: end for
25: Output $x_{optimal}$ and corresponding Map
```

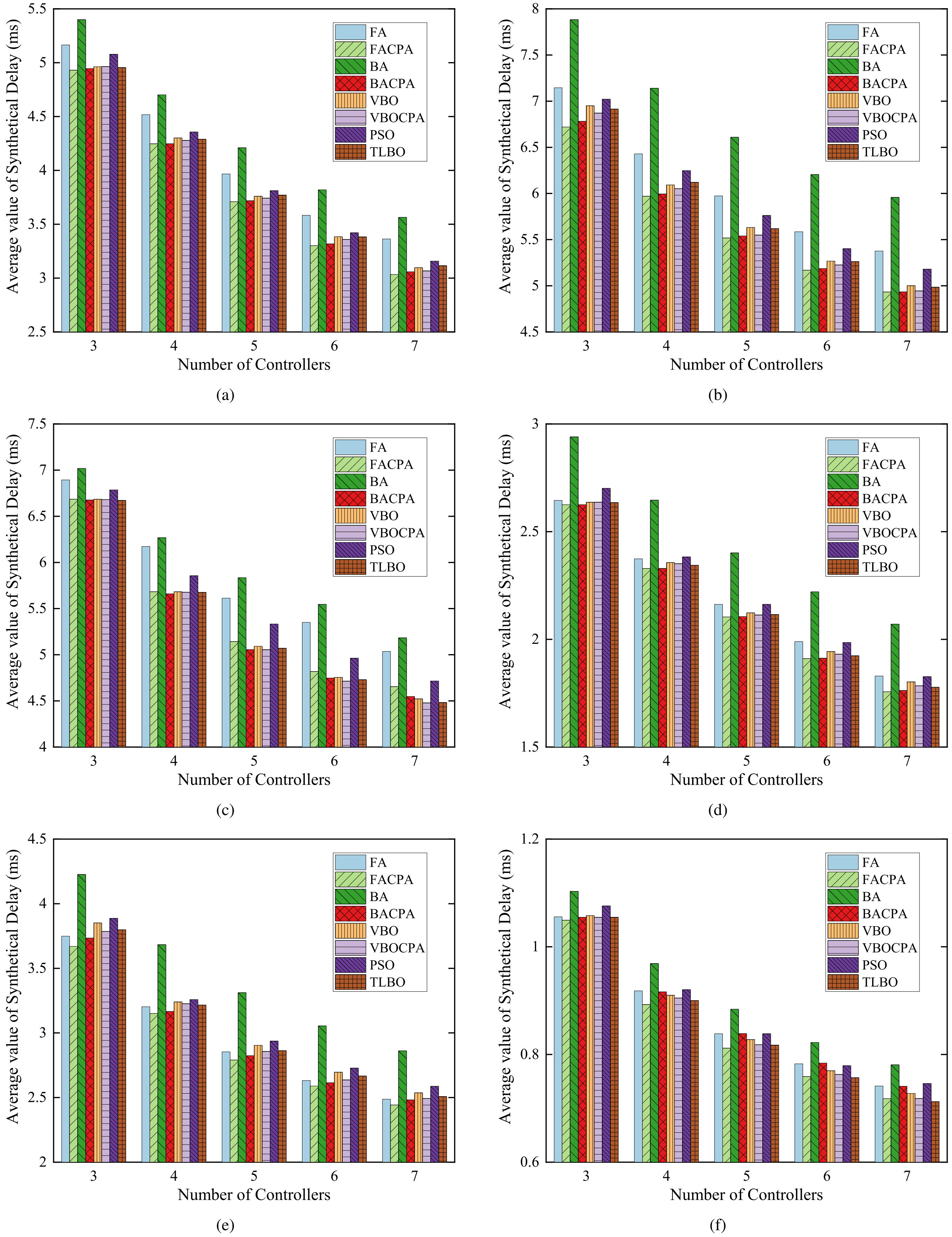
D. PSO-BASED PARAMETER OPTIMIZATION ALGORITHM

Additionally, for the parameter optimization problem of the Controller Placement Algorithm (CPA), we designed the parameter optimization algorithm based on the PSO. The PSO was a heuristic algorithm, which originated from the ethnic behavior of birds to transfer foraging information [9]. The execution process of the PSO-based parameter optimization algorithm (PSOPOA) is shown in Algorithm 4.

Firstly, generate p sets of key parameters of the CPA randomly (lines 1 to 4). Then, iteratively optimize the entire solution space (lines 6 to 11). At last, output the optimal parameters of the CPA (line 12). rand (0, 1) in Algorithm 2 means that random numbers between 0 and 1 are taken. For the description of the key formulas in the PSO, refer to [9].

IV. SIMULATION AND EVALUATION

We used different heuristic algorithms to solve the SDCPM. The synthetical delay of real topologies was calculated according to the precise location of the controllers and the mapping relationship.



(a)

(b)

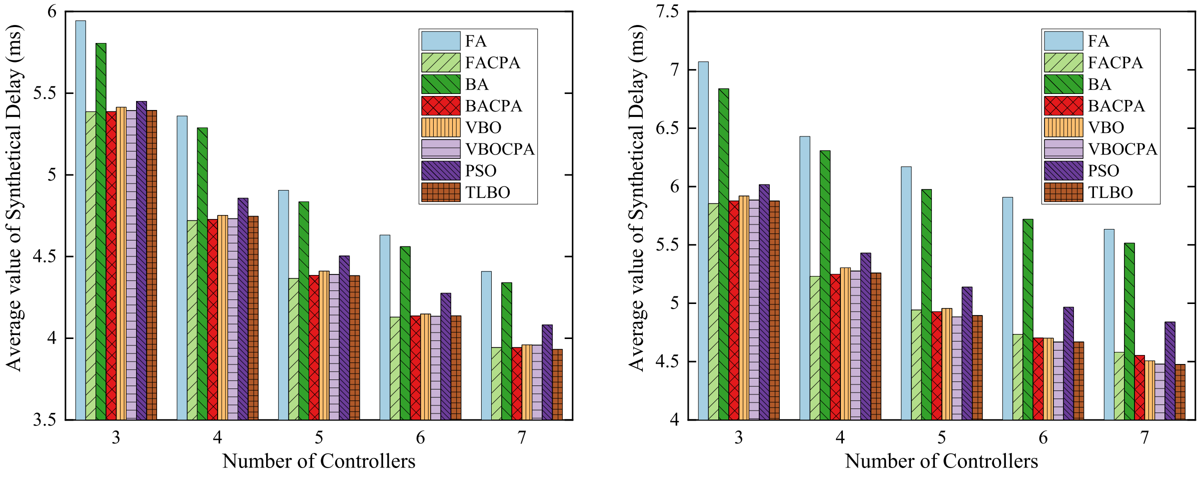
(c)

(d)

(e)

(f)

FIGURE 2. Average value of DELAY of different synthetical-delay controller placement algorithms, (a) Savvis (b) Rnp (c) Ans (d) Bosnet (e) Ernet (f) Funet (g) Internet 2 (h)Bell Canada.



(g)

(h)

FIGURE 2. *(Continued.)* Average value of DELAY of different synthetical-delay controller placement algorithms, (a) Savvis (b) Rnp (c) Ans (d) Bosnet (e) Ernet (f) Funet (g) Internet 2 (h)Bell Canada.

A. SIMULATION SETTINGS

1) BASIC ENVIRONMENT

The development environment was windows 10 professional (64 bit), intel(R) Core (TM) i5-7300HQ CPU@2.50GHz, and 16.00G memory. All solving algorithms and models were implemented by using the Python programming language.

Algorithm 4 PSOPOA

```
Input: iteration times: iter, number of particles: p, topology
    information: G, initial parameters variable of PSO: $c1$ and $c2$
Output: $global_{bestcoor}$
Randomly generate vector $x$ and vector $v$
According to (12), use p sets of $x$ as the input of the CPA,
    and obtain p result values
Calculate $personal_{bestcoor}$ and $personal_{best}$
Calculate $global_{bestcoor}$, and $global_{best}$
for t in (0, iter) do
    $v = (0.4 + 0.5*(1 - \text{t/iter}))*v_{old} + c1 * \text{rand(0,1)} * (personal_{bestcoor} - x_{old}) + c2 * \text{rand(0,1)} * (global_{bestcoor} - x_{old})$
    $x = x_{old} + v$
    According to (12), calculate p values of the result
    Calculate $personal_{bestcoor}$ and $personal_{best}$
    Calculate $global_{bestcoor}$, and $global_{best}$
end for
Output $global_{bestcoor}$
```

2) TOPOLOGIES

TABLE 1. Information of topologies.

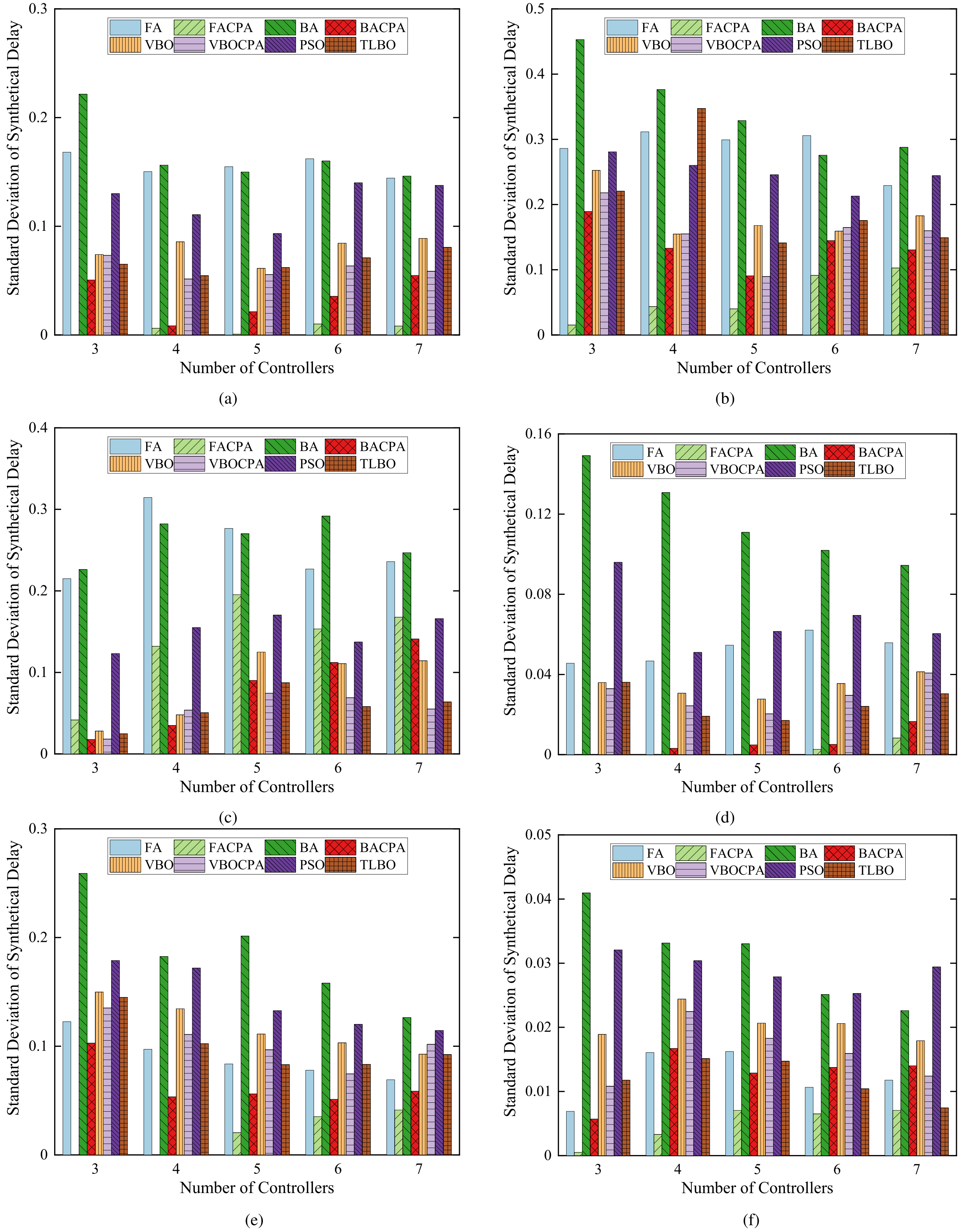
Network topologies	Number of nodes	Number of links
Savvis	19	20
Rnp	28	31
Ans	18	25
Bsonet	14	19
Ernet	16	16
Funet	24	27
Internet 2	36	45
Bell Canada	48	66

The 7 random WAN topologies from The Internet Topology Zoo and the topology of Internet2 Network Advanced Layer 2/3 Service were selected as topology instances [42], [43]. Distance between nodes was calculated according to the longitude and latitude of the nodes provided by [42], [43]. The unit is m. Considering that the delay of data packets includes propagation delay, processing delay, and transmission delay, the data transmission speed on the link was set to two-thirds of the speed of light, $2 \times 10^8 m/s$ [33]–[35]. The relevant parameters of the 8 topologies are shown in Table 1.

3) COMPARATIVE ALGORITHMS

The PSO in [18], the TLBO in [20], the FA in [25], the VBO in [29], and the BA with unoptimized parameters were compared with the parameter-optimized BACPA, FACPA, and VBOCPA. It should be noted that the PSO can be applied not only to the SDCPM, but also to the POM. To distinguish, the algorithm used in [18] to solve the CPP is referred to as PSO in this article, and the PSO applied to solve the POM is referred to as PSOPOA. Similarly, the algorithms with optimized parameters is denoted as BACPA, FACPA, and VBOCPA in this article, and the algorithms with unoptimized parameters is denoted as BA, FA, and VBO. The settings of the relevant parameters of the algorithms used in the simulation are shown in Table 2.

The parameter symbols of the above comparison algorithm are the original symbols in relevant literatures. The parameters of the BACPA, the FACPA, and the VBOCPA are the optimized results of the PSOPOA. The parameters of the PSO, the BA, the FA, the TLBO, and the VBO are the original values of [9]–[12], and [13]. It should be noted that



(a)

(b)

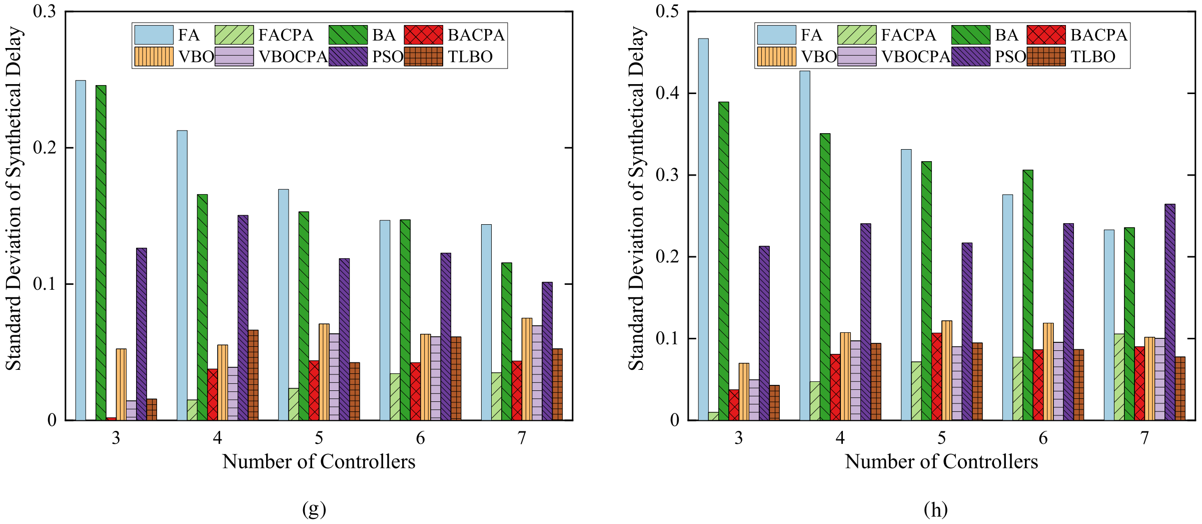
(c)

(d)

(e)

(f)

FIGURE 3. **Standard deviation of DELAY of different synthetical-delay controller placement algorithms, (a) Savvis (b) Rnp (c) Ans (d) Bosnet (e) Ernet (f) Funet (g) Internet 2 (h)Bell Canada.**



(g)

(h)

FIGURE 3. ***(Continued.)*** **Standard deviation of DELAY of different synthetical-delay controller placement algorithms, (a) Savvis (b) Rnp (c) Ans (d) Bosnet (e) Ernet (f) Funet (g) Internet 2 (h)Bell Canada.**

TABLE 2. **Parameter settings of algorithms.**

Algorithms	T	n	Related parameters
FA	100	20	α=0.2, r=1, β_0=1, ε=Norm(0,1)
FACPA	100	20	α=2.62, r=0.29, β_0=44.16, ε=Norm(0,1)
BA	100	20	$f_{\min}$=0, $f_{\max}$=1, β=0.9, γ=0.9, R_0=0.6, A_0=0.6
BACPA	100	20	$f_{\min}$=-1.04, $f_{\max}$=0.58, β=1.09, γ=0.01, R_0=2.46, A_0=4.67
VBO	100	20	α=0.1, $c1$=1.5, $c2$=1.25
VBOCPA	100	20	α=0.32, $c1$=8.71, $c2$=2.81
PSO	100	20	ω=0.4+0.5*(1-t/T), $c1$=2, $c2$=2
TLBO	100	20	T_F=randint(1,2)
PSOPOA	50	20	ω=0.4+0.5*(1-t/T), $c1$=2, $c2$=2

T represents the number of iterations of the algorithm, and n represents the population size (i.e., the number of particles, the number of bats, the number of agents…). Norm(0,1) represents a random number of Gaussian distributions with an expectation of zero and a standard deviation of one. randint(0,1) represents a random integer from 1 to 2 (either 1 or 2). $\omega = 0.4 + 0.5 * (1 - \mathrm{t/T})$ is an improvement of the PSO, which introduce the iteration-based dynamic strategy, where t represents the current quantity of iterations.

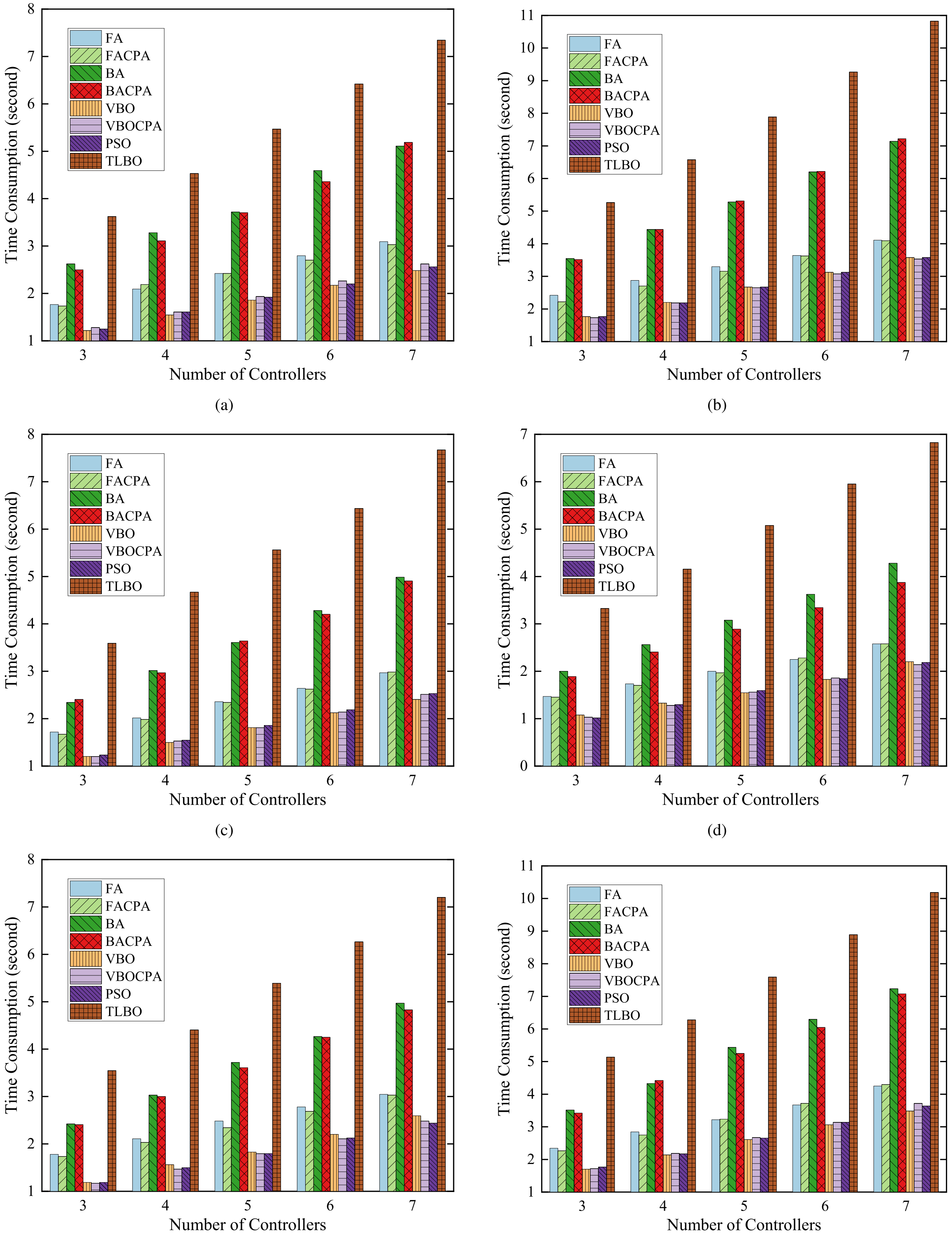
4) METRICS

To place 3 to 7 controllers in 8 topologies respectively, then, we run the FA, the FACPA, the BA, the BACPA, the VBO, the VBOCPA, the PSO, and the TLBO 100 times each, and recorded the results of 100 sets of the DELAY. To take the delay between the controllers and the switches as the main optimization indicator, we set the weight coefficient (λ_1) of the delay between the controllers and the switches to 0.8, and the weight coefficient (λ_2) of the delay between the controllers to 0.2. We took the mean and variance of the 100 sets of the DELAY as metric 1 and metric 2. With the different numbers of controllers placed in the 8 topologies, we took the respective time consumption of the FA, the FACPA, the BA, the BACPA, the VBO, the VBOCPA, the PSO, and the TLBO as metric 3.

B. SIMULATION ANALYSIS

As shown in Fig. 2, no matter what algorithm is used, the DELAY in the 8 topologies decreases with the increase of the number of controllers. Because in the synthetical delay indicator used in the simulation, the delay weight coefficient between the controllers is only 0.2, and the delay between the controllers and the switches still occupies the main part. With the increase of the number of the controllers, each switch can be assigned to a controller that is closer to itself. Therefore, the value of DELAY decreases as the number of controllers increases.

It can also be seen from Fig. 2 that in the 8 topologies, the results of the TLBO are lower than those of the PSO. The results of FA with unoptimized parameters are higher than those of the PSO in 5 topologies of Savvis (Fig. 2a), Rnp (Fig. 2b), Ans (Fig. 2c), Internet 2 (Fig. 2g), and Bell Canada (Fig. 2h). While the results of the optimized FACPA are lower than those of the PSO in 8 topologies. Furthermore, the results of FACPA are lower than those of TLBO in 4 topologies of Savvis (Fig. 2a), Rnp (Fig. 2b), Bosnet (Fig. 2d), and Ernet (Fig. 2e). In the 8 topologies, the results of BA with unoptimized parameters are higher than those of PSO. While the results of optimized BACPA are lower than those of the PSO. Furthermore, the results of the BACPA are lower than those of the TLBO in 4 topologies of Savvis (Fig. 2a), Rnp (Fig. 2b), Bosnet (Fig. 2d), and Ernet (Fig. 2e). In the 8 topologies, the VBO with unoptimized parameters are lower than those of the PSO and are higher than those of the TLBO. While in the 3 topologies of Savvis (Fig. 2a), Rnp (Fig. 2b), and Ans (Fig. 2c), the results of the optimized VBOCPA are lower



(a)

(b)

(c)

(d)

(e)

(f)

FIGURE 4. Time consumption of different synthetical-delay controller placement algorithms, (a) Savvis (b) Rnp (c) Ans (d) Bosnet (e) Ernet (f) Funet (g) Internet 2 (h)Bell Canada.

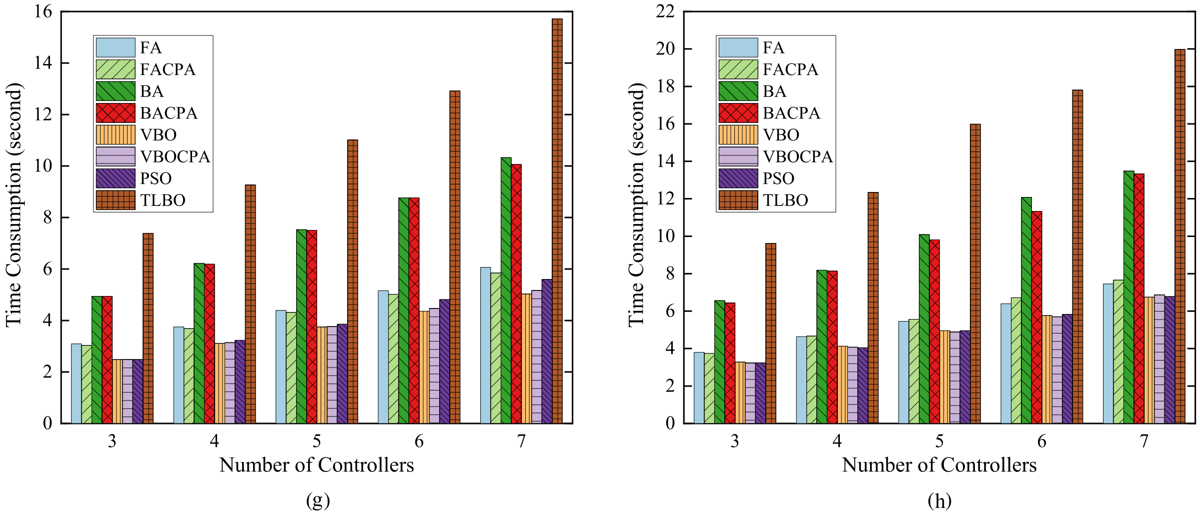


FIGURE 4. ***(Continued.)*** **Time consumption of different synthetical-delay controller placement algorithms, (a) Savvis (b) Rnp (c) Ans (d) Bosnet (e) Ernet (f) Funet (g) Internet 2 (h)Bell Canada.**

than those of the TLBO. In the 8 topologies, the results of the FACPA, the BACPA, and the VBOCPA are lower than those of the original algorithms to varying degrees. Therefore, the synthetical delay of the controller placement scheme solved by the FACPA, the BACPA, and the VBOCPA is lower.

It can be seen from Fig. 3 that in the 8 topologies, the results of TLBO are lower than those of PSO. In the 8 topologies, compared with the results of FA with unoptimized parameters, the results of the optimized FACPA are evidently lower. Except for Bell Canada (Fig. 3h), the results of the FACPA are lower than those of the TLBO. In the 8 topologies, compared with the results of BA with unoptimized parameters, the results of the optimized BACPA are much lower. Except for Funet (Fig. 3f) and Bell Canada (Fig. 3h), the results of the BACPA are lower than those of the TLBO. In the 8 topologies, compared with the results of VBO with unoptimized parameters, the results of the optimized VBOCPA are much lower. In the topologies of Savvis (Fig. 3a) and Ans (Fig. 3c), the results of the VBOCPA are lower than those of the TLBO. Therefore, the synthetical delay of the controller placement scheme obtained by the FACPA, the BACPA, and the VBOCPA is more stable.

As can be seen from Fig. 4, regardless of the adopted algorithm, the time consumption in the 8 topologies increases with the number of placed controllers. This is because when more controllers are placed, the dimension of the problem may increase. In the optimized algorithms and the original algorithms, only a few key parameters are changed, and the main body of the algorithms have not been changed, so the time consumption of the optimized algorithms is not significantly different from that of the original algorithms. The time consumption of the TLBO is much higher than that of the other algorithms, while the time consumption of the PSO is comparable to that of the VBO and the VBOCPA.

The synthetical delay of the controller placement schemes solved by the FACPA, the BACPA, and the VBOCPA is lower than that of the PSO and the original algorithms and is comparable to that of the TLBO. The variance of the synthetical delay of the controller placement schemes solved by the FACPA, the BACPA, and the VBOCPA is lower than that of the PSO, the TLBO, and the original algorithms. The time consumption of the FACPA, the BACPA, and the VBOCPA is comparable to that of the original algorithms and is sharply lower than that of the TLBO. To sum up, the better performance of the FACPA, the BACPA, and the VBOCPA verifies the effectiveness of the POM.

V. CONCLUSION

Currently, the heuristic algorithm, which is of low-time complexity and is suitable for solving high-dimensional problems, is the main algorithm for solving the CPP. However, if the parameters of the heuristic algorithm are improperly set, it will have a negative effect on the results. In response to this problem, we established the POM. To verify the effectiveness of the POM, we established the SDCPM, and respectively designed the PSOPOA, the FACPA, the BACPA, and the VBOCPA, based on the PSO, the FA, the BA, and the VBO. Simulations were conducted in the 8 real topologies. The results show that compared with the original algorithms, in terms of the synthetical delay, the variance of synthetical delay, and the time consumption, the TLBO, and the PSO, the FACPA, the BACPA, and the VBOCPA perform better. This boils down to the POM. Besides, the POM can also be used to optimize the parameters of other heuristic algorithms for solving the CPP. In future work, we will further study the parameter optimization problem of the heuristic algorithm for the CPP containing more network indicators to obtain a reasonable SDN controller placement scheme.

REFERENCES

[1] H. Farhady, H. Lee, and A. Nakao, "Software-defined networking: A survey," *Comput. Netw.*, vol. 81, pp. 79–95, Apr. 2015.

[2] T. Hu, Z. Guo, P. Yi, T. Baker, and J. Lan, "Multi-controller based software-defined networking: A survey," *IEEE Access*, vol. 6, pp. 15980–15996, 2018.

[3] G. Wang, Y. Zhao, J. Huang, and W. Wang, "The controller placement problem in software defined networking: A survey," *IEEE Netw.*, vol. 31, no. 5, pp. 21–27, Sep. 2017.

[4] B. P. R. Killi and S. V. Rao, "Controller placement in software defined networks: A comprehensive survey," *Comput. Netw.*, vol. 163, Nov. 2019, Art. no. 106883.

[5] B. Heller, R. Sherwood, and N. McKeown, "The controller placement problem," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 4, pp. 473–478, Sep. 2012.

[6] D. Hock, M. Hartmann, S. Gebert, M. Jarschel, T. Zinner, and P. Tran-Gia, "Pareto-optimal resilient controller placement in SDN-based core networks," in *Proc. 25th Int. Teletraffic Congr. (ITC)*, Sep. 2013, pp. 1–9.

[7] A. Gogna and A. Tayal, "Metaheuristics: Review and application," *J. Exp. Theor. Artif. Intell.*, vol. 25, no. 4, pp. 503–526, 2013.

[8] Z. Beheshti and S. M. H. Shamsuddin, "A review of population-based meta-heuristic algorithms," *Int. J. Adv. Soft Comput. Appl*, vol. 5, no. 1, pp. 1–35, 2013.

[9] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci. (MHS)*, 1995, pp. 39–43.

[10] X. S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization*. Berlin, Germany: Springer, 2010, pp. 65–74.

[11] X-S. Yang, "Firefly algorithm, stochastic test functions and design optimisation," *Int. J. Bio-Inspired Comput.*, vol. 2, no. 2, pp. 78–84, 2010.

[12] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems," *Comput.-Aided Des.*, vol. 43, no. 3, pp. 303–305, 2011.

[13] A. K. Singh and S. Srivastava, "Varna-based optimization: A new method for solving global optimization," *Int. J. Intell. Syst. Appl.*, vol. 10, no. 12, pp. 1–15, Dec. 2018.

[14] M. Tanha, D. Sajjadi, R. Ruby, and J. Pan, "Capacity-aware and delay-guaranteed resilient controller placement for software-defined WANs," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 3, pp. 991–1005, Sep. 2018.

[15] S. Lange, S. Gebert, T. Zinner, P. Tran-Gia, D. Hock, M. Jarschel, and M. Hoffmann, "Heuristic approaches to the controller placement problem in large scale SDN networks," *IEEE Trans. Netw. Service Manage.*, vol. 12, no. 1, pp. 4–17, Mar. 2015.

[16] J. Liao, H. Sun, J. Wang, Q. Qi, K. Li, and T. Li, "Density cluster based approach for controller placement problem in large-scale software defined networkings," *Comput. Netw.*, vol. 112, pp. 24–35, Jan. 2017.

[17] V. Ahmadi and M. Khorramizadeh, "An adaptive heuristic for multi-objective controller placement in software-defined networks," *Comput. Electr. Eng.*, vol. 66, pp. 204–228, Feb. 2018.

[18] C. Gao, H. Wang, F. Zhu, L. Zhai, and S. Yi, "A particle swarm optimization algorithm for controller placement problem in software defined network," in *Proc. Int. Conf. Algorithms Archit. Parallel Process.* Cham, Switzerland: Springer, 2015, pp. 44–54.

[19] A. Jalili and M. Keshtgari, "A new reliable controller placement model for software-defined WANs," *J. Data Mining*, vol. 8, no. 2, pp. 269–277, 2020.

[20] A. K. Singh, S. Maurya, N. Kumar, and S. Srivastava, "Heuristic approaches for the reliable SDN controller placement problem," *Trans. Emerg. Telecommun. Technol.*, vol. 31, no. 2, p. e3761, Feb. 2020.

[21] M. Khorramizadeh and V. Ahmadi, "Capacity and load-aware software-defined network controller placement in heterogeneous environments," *Comput. Commun.*, vol. 129, pp. 226–247, Sep. 2018.

[22] A. Jalili, M. Keshtgari, and R. Akbari, "Optimal controller placement in large scale software defined networks based on modified NSGA-II," *Applied Intell.*, vol. 48, no. 9, pp. 2809–2823, Sep. 2018.

[23] R. Chai, Q. Yuan, L. Zhu, and Q. Chen, "Control plane delay minimization-based capacitated controller placement algorithm for SDN," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, pp. 1–17, Dec. 2019.

[24] Z. Zhao and B. Wu, "Scalable SDN architecture with distributed placement of controllers for WAN," *Concurrency Comput., Pract. Exper.*, vol. 29, no. 16, p. e4030, Aug. 2017.

[25] K. Sagar Sahoo, A. Sarkar, S. K. Mishra, B. Sahoo, D. Puthal, M. S. Obaidat, and B. Sadun, "Metaheuristic solutions for solving controller placement problem in SDN-based WAN architecture," in *Proc. 14th Int. Joint Conf. e-Bus. Telecommun.*, 2017, pp. 1–10.

[26] G. Wang, Y. Zhao, J. Huang, and Y. Wu, "An effective approach to controller placement in software defined wide area networks," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 1, pp. 344–355, Mar. 2018.

[27] A. Jalili, M. Keshtgari, and R. Akbari, "A new framework for reliable control placement in software-defined networks based on multi-criteria clustering approach," *Soft Comput.*, vol. 24, no. 4, pp. 2897–2916, Feb. 2020.

[28] F. J. Ros and P. M. Ruiz, "On reliable controller placements in software-defined networks," *Comput. Commun.*, vol. 77, pp. 41–51, Mar. 2016.

[29] A. K. Singh, S. Maurya, and S. Srivastava, "Varna-based optimization: A novel method for capacitated controller placement problem in SDN," *Frontiers Comput. Sci.*, vol. 14, no. 3, Jun. 2020, Art. no. 143402.

[30] B. Zhang, X. Wang, and M. Huang, "Multi-objective optimization controller placement problem in Internet-oriented software defined network," *Comput. Commun.*, vol. 123, pp. 24–35, Jun. 2018.

[31] P. Xiao, Z.-Y. Li, S. Guo, H. Qi, W.-Y. Qu, and H.-S. Yu, "A K self-adaptive SDN controller placement for wide area networks," *Frontiers Inf. Technol. Electron. Eng.*, vol. 17, no. 7, pp. 620–633, Jul. 2016.

[32] Y. Hu, T. Luo, N. C. Beaulieu, and C. Deng, "The energy-aware controller placement problem in software defined networks," *IEEE Commun. Lett.*, vol. 21, no. 4, pp. 741–744, Apr. 2017.

[33] J. Zhao, H. Qu, J. Zhao, Z. Luan, and Y. Guo, "Towards controller placement problem for software-defined network using affinity propagation," *Electron. Lett.*, vol. 53, no. 14, pp. 928–929, Jul. 2017.

[34] W. Chen, C. Chen, X. Jiang, and L. Liu, "Multi-controller placement towards SDN based on Louvain heuristic algorithm," *IEEE Access*, vol. 6, pp. 49486–49497, 2018.

[35] M. J. F. Alenazi and E. K. Çetinkaya, "Resilient placement of SDN controllers exploiting disjoint paths," *Trans. Emerg. Telecommun. Technol.*, vol. 31, no. 2, p. e3725, Feb. 2020.

[36] G. Li, X. Wang, and Z. Zhang, "SDN-based load balancing scheme for multi-controller deployment," *IEEE Access*, vol. 7, pp. 39612–39622, 2019.

[37] J. Liu, J. Liu, and R. Xie, "Reliability-based controller placement algorithm in software defined networking," *Comput. Sci. Inf. Syst.*, vol. 13, no. 2, pp. 547–560, 2016.

[38] A. Jalili, M. Keshtgari, R. Akbari, and R. Javidan, "Multi criteria analysis of controller placement problem in software defined networks," *Comput. Commun.*, vol. 133, pp. 115–128, Jan. 2019.

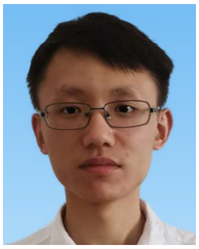
[39] B. P. R. Killi and S. V. Rao, "Optimal model for failure foresight capacitated controller placement in software-defined networks," *IEEE Commun. Lett.*, vol. 20, no. 6, pp. 1108–1111, Jun. 2016.

[40] G. Yao, J. Bi, Y. Li, and L. Guo, "On the capacitated controller placement problem in software defined networks," *IEEE Commun. Lett.*, vol. 18, no. 8, pp. 1339–1342, Aug. 2014.

[41] F. A. Ozsoy and M. C. Pinar, "An exact algorithm for the capacitated vertex -center problem," *Comput. Oper. Res.*, vol. 33, no. 5, pp. 1420–1436, May 2006.

[42] S. Knight, H. X. Nguyen, N. Falkner, R. Bowden, and M. Roughan, "The Internet topology zoo," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 9, pp. 1765–1775, Oct. 2011.

[43] *Internet2 Open Science, Scholarship and Services Exchange*. Accessed: Jul. 2020. [Online]. Available: https://www.internet2.edu/products-services/advanced-networking/layer-2-services/



YI LI is currently pursuing the master's degree with the School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai, China. His current research interests include big data, blockchain, networks, and communications.

SHAOPENG GUAN is currently an Associate Professor with the School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai, China. His current research interests include big data, blockchain, networks, and communications.

CONGHUI ZHANG is currently pursuing the master's degree with the School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai, China. His current research interests include big data, blockchain, networks, and communications.

WENWEN SUN is currently pursuing the master's degree with the School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai, China. Her current research interests include big data, blockchain, and networks, and communications.