An Enhanced Forwarding Method based on Intelligent Water Drops Algorithm in Named Data Network

HamidReza Afzal[1], Behrang Barekatain[1,2*], Zahra Beheshti[1,2]
1- Department of Computer Engineering, Najafabad Branch, Islamic Azad university, Najafabad, Iran.
2- Big Data Research Center, Najafabad Branch, Islamic Azad university, Najafabad, Iran.
Email: Behrang_Barekatain@iaun.ac.ir (Corresponding author)

Received: April 2021 Revised: July 2021 Accepted: August 2021

ABSTRACT:
Although Named Data Network (NDN) has made a bright future in the Internet for high volume of requests by many users, how to send a request package (I-Pkt) consciously from the consumer to the producer and returning the data package (D-Pkt) inversely is still one of its most important challenges. According to the recent limited researches, using service quality parameters beside an optimization algorithm like ant colony to find the optimal path has been an appropriate response to solve this problem. Insufficient attention to service quality parameters to find an optimal path is a problem to be focused on. In line with this, this study has addressed forwarding using the Intelligent Water Drops (IWD) optimization algorithm. In this algorithm, the best possible path based on many parameters related to path quality like cost, delay and number of steps taken by the package (I-Pkt) will be selected. In other words, the water drop movement tries to find the optimal path through the amount of soil collection in the path or the velocity rate. The results of simulating in ndnSIM simulator show that comparing the ant colony, it has been improved in cost parameter on average by 23%, in hit ratio parameter by 24% and in hop count parameter by 13%. The general result is increasing service level quality in NDNs.

KEYWORDS: Named Data Network, Forwarding, Intelligent Water Drop Algorithm, Cost, Hit Ratio, Hop Count.

1. INTRODUCTION

Nowadays, the base of information forwarding and receiving by many users is content and disregarding the point if they know where to reach the target information in the network, they would like to know how to access the information they need. This shows the necessity of change in the present structure of global Internet network in near future. As the best possible response to this request, the Named Data Networks (NDNs) were proposed and despite IP-based networks, no final destination is considered in them [1]. In these networks, the content names are used instead of source and destination IP addresses [2]. Using the phrase “Internet Future” for this type of network shows the significance and effect of them in a very near future. Although NDN gives the hope to solve some problems of current users of Internet network, it faces many challenges including naming, saving information in caching, mobility, security, applications, routing and forwarding [3].

According to the last issue, the method of Interest I-Pkt forwarding from consumer to producer and returning the Data D-Pkt in the inverse path is one of its most significant challenges [4]. In other words, after the consumer sent the I-Pkt, the existence of an effective method for forwarding it towards the destination (producer) consciously will have an important effect in the efficiency of data exchange method. Not considering the network current conditions and using only one path can cause the I-Pkt not to reach the producer at the right time. After using service quality parameters beside each optimization algorithm, we can give it more capability to find the shortest optimal path.

To solve this problem and regarding the fact that the subject is new, very limited numbers of algorithms have been proposed so far to send the packet in this kind of networks. For example, some has used ant colony optimization algorithm to send the I-Pkt to the producer [5]. Some other researchers have used ant colony optimization algorithm accompanying the use of service quality parameter to find the optimal path [6, 7]. Some has used service quality parameter only to find the shorter path (not the optimal one) [8]. In another method, SDN network concepts and combination of its concepts have been used to provide a forwarding method [9]. Another method has also used Q-learning concept in forwarding the I-Pkt towards the producer [10]. One other method tried to use interfaces to send the packet through ranking the output interfaces

Paper type: Research paper
DOI: https://doi.org/10.52547/mjee.15.4.77
How to cite this paper: H. R. Afzal, B. Barekatain and Z. Beheshti, **“An Enhanced Forwarding Method based on Intelligent Water Drops Algorithm in Named Data Network”**, *Majlesi Journal of Electrical Engineering*, Vol. 15, No. 4, pp. 77-86, 2021.

which has been used less before [11].

According to these researches, it seems that still no effective method that can consider the network current conditions based on important and different parameters of service quality and also the issue of finding the optimal path not necessarily the short one in forwarding the I-Pkt has been provided.

Therefore, this study has tried to consider the most important service quality parameters including Delay, Hop Count, Hit Ratio and Cost in finding the optimal path. Intelligent Water Drops (IWD) algorithm which was used in this study is an algorithm which computes the probability of different paths through soil parameter and calculation of packet movement velocity comparing the other optimization algorithms like ant colony. In this algorithm, the intelligent water drops during their movement firstly have the capability to increase or decrease the amount of path soil and secondly, they also use a concept called velocity in forwarding the packet. The results of the proposed method simulation show that comparing the most recent provided method (ant colony) related to this study, it has improved the cost, hit ratio, delay and hop count on average by 20, 23, 24 and 13 percent, respectively.

The rest of this paper is organized as follows. In section 2, the related work is discussed in details. Section 3 presents the preliminary definitions. Then in section 4, the proposed method will be provided. In section 5, simulation results are assessed and finally, in section 6, the conclusion will be presented.

2. RELATED WORK

One of important classifications in I-Pkt forwarding methods in NDNs is comparative packet forwarding method. It means that using the current state information to send the I-Pkt in the network, the routers can compute the output interfaces based on significant parameters in delivering the packet like the error rate of links, network congestion, using several paths and maintaining the current state. The studies which have worked in this field are as follows.

In the article by Xiao et al. [11], a method called Rank-Based Routing Strategy (RBRS) was provided which ranks each interface for routing selection. In FIB and PIT tables, a column called outface was added to rank the output interfaces in receiving D-Pkt. The higher of interface rank (lower numerical amount) is used to send the I-Pkt. This method has not been examined in a more complicated scenario with more nodes and more I-Pkts.

In the method used by Li et al. [12], Nack I-Pkt has been used as a tool for the first time in response to the requested packets of which its corresponding data packets will not be opened as of any reasons. Also, interface color coding has been used to send the I-Pkt towards them. This method emphasizes more on the packets which do not reach the destination and no specific algorithm in forwarding the packet towards the destination has been examined.

In [6], this point was mentioned that regarding the existence of different methods in forwarding the packet, most of these methods do not use service quality parameters while finding the path toward producer. Considering the fact that ant colony optimization algorithm is only able to find the shortest path, this path is not necessarily optimized, because the network current state like links state, network congestion and cost are not considered.

In the method provided by Crooch et al. [7], this point was mentioned that NDN uses a symmetrical packet routing i.e. the request packets are received from a path with determined number of forwarding and its related data packets are received from the same path but in inverse direction. The original idea in this study is that the ANTs which are the packets sent in the network can measure service quality parameters and then, the desired pheromone amount will be distributed in the path based on that.

In an article presented in 2019 [9], combination of SDN[1] and NDN network concepts has been used to make a better method to save the information in caching and faster packet forwarding to the next routers. The advantage of this method comparing the previous ones is as follows: combining the current structures and concepts in SDN and NDN and making a new method of placing the data on the data stored in caching of routers. This has caused the operation of packet forwarding towards next routers become faster in addition to more optimized data savings.

The advantage of this approach to the previous methods is that, with the combination of the structure and concepts' existing in SDN aperture, a new method of placing the data on the data stored in caching of routers is made. In addition, to more efficient storage of data, packet transmission is sent to the next routers as well. The base of this method is not optimal path method i.e. there may be a faster but longer path to send the packet. Also, in [13], a packet forwarding method combined NDN and VANET[2] networks to prevent a storm in forwarding high volume and delay in reaching the destination.

To prevent the congestion in forwarding the I-Pkt, in [14], a probability method has been used to compute interface congestion control and then, forwarding the packet through an optimized interface. In [15], a packet forwarding method has been used in which through sharing the information of a router with a request

[1] Software Define Network

[2] Vehicular Ad-hoc Network

packet, the same is done for the other request packets. This method causes better performance in not having unwanted congestion in the network and effective packet restore (D-Pkt) from adjacent nodes comparing the current methods of packet forwarding in NDNs.

Q-learning is a concept that has been used in [10] to minimize the delay in forwarding the packet from the source to the destination. In this method, using Q-learning and two phases of exploration and exploitation in which the first phase is collecting the path information and the second phase is forwarding the packet towards the next node reduces the amount of delay in forwarding the packet. This method has only concentrated on one parameter to minimize it and the other parameters have not been examined.

When the I-Pkt is sent from a consumer to the producer, usually the size of the D-Pkt in response to that is larger. This causes facing the reception of high volume of packets (D-Pkt) in case of forwarding several packets (I-Pkt) simultaneously. This challenge mentioned in [16] aims at assessing the output interfaces using fuzzy logic. As a result, by forwarding the I-Pkt through the interface, it will have the least congestion control and less packet loss rate.

Cache Storage (CS) table is the same as router's caching, Pending Interest Table (PIT) is devoted to keep the belonged request packet and Forwarding Information Base (FIB) table is devoted to keep the request packets which are waiting to be sent to the other routers. When the content of a I-Pkt in CS and PIT cannot be found, it is referred to FIB.

In [17], a Forwarding strategy based on Recommendation Algorithm (FRA) has been presented in which packet forwarding (I-Pkt) from FIB to other routers is done based on Forwarding Algorithm based on Packet Recommendation (FAPR) or Forwarding Algorithm based on Node Recommendation (FANR). In the first method, receiving an I-Pkt in the node FIB and saving the information, all the request packets similar to this one will receive a similar suggestion to be sent. In the second method, saving the forwarding information of side nodes, it is tried to recommend these nodes to make the request packets pass them.

According to the examination of the aforementioned methods, the challenge of current methods in packet forwarding in NDNs can be categorized as follows. The issue of service quality parameters in packet forwarding is a challenge which has not been considered in recent studies. This is an important point because of packet movement in this kind of networks and the current network state in packet forwarding. Using an optimization algorithm does not find the optimal path necessarily as packet forwarding in a network follows other parameters like congestion, traffic, etc.; therefore, the necessity of combining these algorithms with service quality parameters is still important and inevitable.

3. PROPOSED METHOD

3.1. Basic Concepts of the Proposed Method based on the IWD Algorithm

The water drops in the river always move from the top (source) to the bottom (destination) according to the law of gravity and they move in this path from the shortest possible path. Every water drop moving in this path carries an amount of river soil with it and this amount is removed from the river soil and added to the water drop. The velocity has a significant role in soil transfer by the water drop in the river. Assume that two water drops are moving from a source towards a destination. When the water drop with higher velocity reaches the destination, it carries more soil comparing the water drop with lower velocity. However, the water drop velocity also changes by the river in a way that water drop velocity in a path with less soil is higher. How to select an optimal path among several ones by the intelligent water drops depends on the amount of soil in the path i.e. a path with more amount of soil is less probable to be selected by the intelligent water drop.

The pseudocode of intelligent water drops algorithm has been shown in "Fig. 1". In this algorithm, first the parameters are set. They are grouped in two categories: 1) static parameters like the number of population, maximum number of iterations and the parameters and coefficients used in the equations, 2) dynamic parameters that are given values per iteration like initial velocity and initial soil for each intelligent water drop will be set to zero. Then, the initial population including water drops is given values in the problem atmosphere.

Table 1 shows the description of all symbols. If an intelligent water drop is in node *i* and wants to move towards node *j* and the amount of soil between these two nodes is ($soil\ (i,j)$), to update the velocity of intelligent water drop ($vel^{IWD}(t)$), equation (1) is used:

$$vel^{IWD}(t+1) = vel^{IWD}(t) + \frac{a_v}{b_v + c_v * soil^2(i,j)} \quad (1)$$

Where, ($vel^{IWD}(t+1)$) is the updated velocity of intelligent water drop in node *j* and a_v, b_v and c_v are the velocity constants which are given in the problem. The time that the intelligent water drop has had to move from node *i* to node *j* with the velocity of ($vel^{IWD}(t+1)$) is computed by equation (2):

$$time(i,j;\ vel^{IWD}) = \frac{HUD(i,j)}{vel^{IWD}} \quad (2)$$

Where, $HUD(i,j)$ is the problem heuristic information in the movement of water drop from node *i* to node j:

$$vel^{IWD}(t+1) = vel^{IWD}(t+1) + \begin{cases} \varepsilon & if\ |vel^{IWD}(t+1)| < \varepsilon \\ 0 & otherwise \end{cases} \quad (3)$$

Table 1. Symbols description.

Symbol	Description
Soil (i,j)	The amount of the soil between node *i* and node *j*
$vel^{IWD}(t)$	The current velocity of the IWD
$vel^{IWD}(t+1)$	The updated velocity of the IWD at the next node *j*
a_v, b_v, and c_v	Constant velocity parameters
$time(i,j;\ vel^{IWD})$	The time taken for an IWD having the velocity vel^{IWD} to move from the current node *i* to its next node *j*
ε, ε_s	Constant parameter with a small positive value
$HUD(i,j)$	The heuristic undesirability of moving from node *i* to node *j*
ρ	A positive numbers
$f\big(soil(i,j)\big)$	The inverse of the soil between node *i* and node *j*
$g\big(soil(i,j)\big)$	To shift the *soil(i,j)* on the path joining nodes *i* and *j* toward positive values
min(.)	Returning the minimum value of its arguments
$vc\ (IWD)$	The nodes that the IWD should not visit to keep satisfied the constraints of the problem
l	Node *l*
T^M	A solution
T^B	Global best solution
N_{IWD}	Number of IWDs
$\Delta soil(i,j)$	The amount of soil carried by IWD from node *i* to node *j*
α, β , γ and δ	Constant values where $\alpha+\beta+\gamma+\delta=1$
P	A prefix
K	An interface

The vel^{IWD} value is obtained from the value of $vel^{IWD}(t+1)$ that causes the vel^{IWD} value not to be zero by determining the value of $\varepsilon = 0.001$.

The amount of soil that moves in the path by the intelligent water drop from node *i* to node *j* is obtained by equation (4):

$$\triangle\, soil(i,j) = \frac{a_s}{b_s + c_s * time^2(i,j;vel^{IWD})}, \quad (4)$$

a_s, b_s and c_s are the soil constants which are given in the problem.

After the intelligent water drop moved from node *i* to node *j*, the amount of soil removed in the path between these two nodes is computed by equation (5):

$$soil(i,j) = (1-\rho) * soil(i,j) - \rho * \Delta soil(i,j) \quad (5)$$

The amount of soil carried by the intelligent water drop from node *i* to node *j* and was added to the intelligent water drop is computed by equation (6):

$$soil^{IWD} = soil^{IWD} + \Delta soil(i,j) \quad (6)$$

Therefore, the movement of intelligent water drop in a path causes path soil reduction and increasing the amount of soil that the intelligent water drop carries.

Pseudo-code of IWD Algorithm

1. Output: Global best solution (T^B)
2. Problem formulation as a graph
3. Initialize the static parameters
4. *Repeat*
5. Initialize dynamic parameters
6. Create IWDs and distribute them on the problem's graph
7. Update the list of visited node
8. *For each IWD do*
    - 8.1 Select the next path of the IWD
    - 8.2 Update the velocity of the IWD
    - 8.3 Compute Δsoil (Eq. 4)
    - 8.4 Remove Δsoil from the path (Eq. 5) and adds it to the IWD (Eq. 6)

9 *End For*

10 Select the best solution in the iteration (T^M)

11 Update the soil value of all edges (Eq. 10)

12 Update the global best solution (T^B) If T^M is better than T^B As: $T^B = T^M$

13 *Until termination condition is satisfied*

14 return (T^B)

Fig.1. The Intelligent Water Drops Algorithm [18].

As it was mentioned, the intelligent water drop selects the path with less soil comparing other paths and the probability of selecting all paths is computed by equation (7):

$$p_i^{IWD}(j) = \frac{f(soil(i,j))}{\sum_{k \notin v_c(IWD)} f(soil(i,k))} \quad (7)$$

Where the function $f(soil(i,j))$ computes the inverse amount of soil between node *i* and node *j* in equation (8):

$$f\big(soil(i,j)\big) = \frac{1}{\varepsilon_s + g\big(soil(i,j)\big)} \quad (8)$$

The constant ε_s is considered a little positive value (0.01) to prevent the denominator to be zero. The function $g\big(soil(i,j)\big)$ causes the change of numerical value of the soil between node *i* and node *j* to a positive value and prevent this value to become negative through equation (9):

$$g\big(soil(i,j)\big) = \begin{cases} soil(i,j) & \text{if} \quad \underset{l \notin vc(IWD)}{min}\big(soil(i,l)\big) \geq 0 \\ soil(i,j) - \underset{l \notin vc(IWD)}{min}\big(soil(i,l)\big) & Otherwise \end{cases} \quad (9)$$

$g\big(soil(i,j)\big)$ is used to shift the *soil(i,j)* on the path joining nodes *i* and *j* toward positive values. The function *min(.)* returns the minimum value of its arguments. The $vc\,(IWD)$ shows the nodes that the IWD should not visit to keep satisfied the constraints of the problem.

Each intelligent water drop moves from the node in which it is placed to the next node to complete its movement path to all nodes. An IWD algorithm iteration finishes when all the intelligent water drops complete their path (T^M), the best solution is selected among them and based on the best solution, the soil of the current path is updated based on equation (10):

$$soil(i,j) = (1-\rho)soil(i,j) + \rho \frac{2 * soil^{IWD}}{N_{IWD}(N_{IWD-1})} \quad \forall(i,j) \in T^M \quad (10)$$

If T^M is better than the best total found path (T^B), T^B will be updated.

3.2. Proposed Method in Details

As packet forwarding in the network must follow the current network state, service quality parameters accompanying an optimization algorithm can be used in selecting an optimal path for I-Pkt forwarding when selecting a forwarding packet. In order to do this, in the proposed method, intelligent water drops have been used. To equalize the proposed method with NDNs, intelligent water drop has been considered as (data, request) packet. The beginning of movement from the source is equal to packet requester. The end of path is equal to the producer and also, the nodes between the paths equal the routers and the passing paths are considered equal to the links. In Table 2, equalization of intelligent water drops algorithm parameters with NDNs have been shown.

Table 2. Adaptation of intelligent water drops algorithm with NDNs.

IWD Algorithm	NDNs
Intelligent water drop	Packet (request-data)
Starting to move from the source	Packet requester
End of the path	Producer or router with caching capability
Nodes between the paths	Routers
Passing paths	links

Fig. 2 shows the flowchart of proposed method after receiving the request I-Pkt by the router. According to this Figure, the algorithm stages after receiving the I-Pkt are as follows:

3-2-1-When the I-Pkt reaches the router, search operation is done to find the name prefix in the D-Pkt in CS.

3-2-2-When the corresponding D-Pkt of this request was found in CS, it moves in the inverse path that the I-Pkt entered until it reaches the consumer and the operation ends at this stage.

3-2-3-If the I-Pkt was not found, the inputs of PIT router (that is the place for saving the request packets' information which are waiting for their corresponding data packets) will be searched for it.

3-2-4-In case that the information of the I-Pkt is in one of the inputs of PIT, it means that similar packets have reached this router by other routers before. Therefore, the number of interface from which this packet came is saved in the input field of PIT and the response of this packet will be received. In case the corresponding D-Pkt reaches, it will be sent through all the interfaces from which this I-Pkt came.

3-2-5-If the I-Pkt is not in the PIT input, it means that no other routers have requested for the packet. Therefore, the search operation will be done in all FIB table inputs to find the corresponding D-Pkt.

3-2-6-If the I-Pkt is not found in FIB table, it will be sent through all router output interfaces to all the adjacent routers or it will be omitted according to the current policies of the router.

3-2-7-In case the I-Pkt is in the FIB table inputs, it means that other routers have requested the response of this packet from this router and therefore, making a new input in PIT table, the packet will be sent towards the adjacent routers to receive the corresponding response.

3-2-1-For all the current interfaces in FIB table, the next path selection probability will be done based on equation (7).

3-2-2-Updating the velocity of intelligent water drop in the path between node *i* and node *j* is computed based on equation (1).

3-2-3-The amount of moved soil by the intelligent water drop in the path between node *i* and node *j* for prefix P is computed by output interface K based on service quality parameters and equation (4) with equation (11).

$$\triangle soil(P,K)_{i,j} = \frac{a_s}{b_s + c_s * time^2(i,j;vel^{IWD})(\alpha * cost + \beta * delay + \gamma * hop\ count + \frac{hit\ ratio}{\delta})} \quad (11)$$

In equation (11), the coefficients α, β , γ and δ are the constants and $\alpha+\beta+\gamma+\delta=1$ i.e. the significance of each service quality parameter in computing the amount of moved soil by the intelligent water drop is the same.

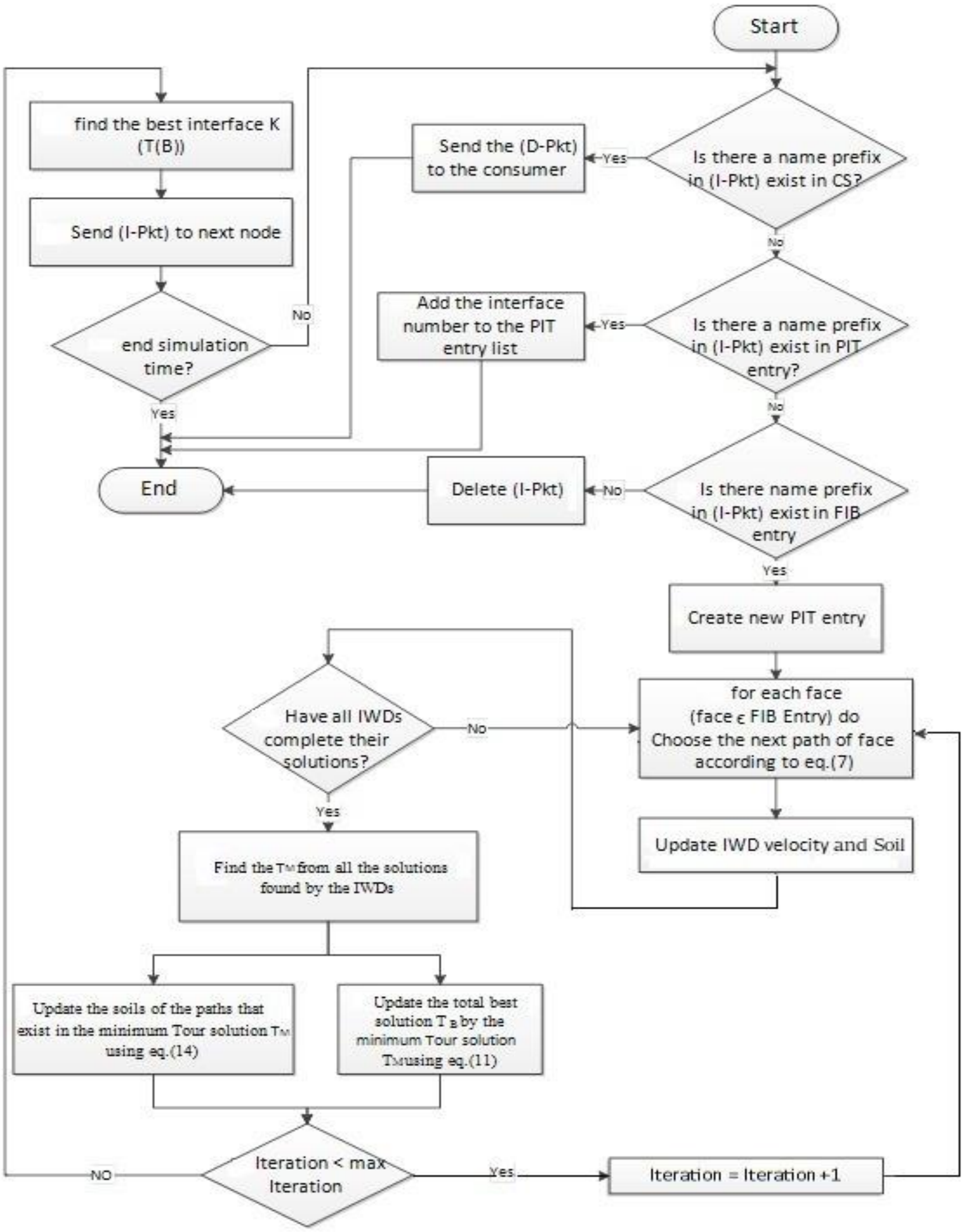

Fig. 2. The flowchart of proposed method after receiving the I-Pkt.

3-2-4-The amount of soil removed from the intelligent water drop from the path is computed using equation (5) for prefix P by interface K through equation (12).

$$soil(P,K)_{i,j} = (1-\rho) * soil(P,K)_{i,j} - \rho * \ \Delta soil(P,K)_{i,j} \quad (12)$$

Also, the amount of soil that is added to intelligent water drop can be computed using equation (6) for prefix P by interface K through equation (13).

$$soil(P,K)^{IWD} = soil(P,K)^{IWD} + \Delta soil(P,K)_{i,j} \quad (13)$$

3-2-5-If all intelligent water drops complete their path, we enter the next stage for computing the completed optimal path by the intelligent water drops; otherwise, we return to step 9 in the flowchart to compute the probability of all output interfaces.

3-2-6-At this stage, the obtained solutions by the intelligent water drops have been examined and the shortest completed path by the intelligent water drop (T^M) is obtained.

3-2-7-The soil of the path obtained at step 9 of the flowchart is updated by equation (14).

$$soil(P,K)_{i,j} = (1-\rho) soil(P,K)_{i,j} + \rho \frac{2 * soil(P,K)^{IWD}}{N_{IWD}(N_{IWD-1})} \quad \forall (i,j) \in T_M \quad (14)$$

3-2-8-In case T^M is better than the obtained global solution (T^B), it will replace the solution.

3-2-9-If the iteration number is less than the maximum number of iteration, we return to step 17 in the flowchart to re-compute the probability of output interfaces.

3-2-10-At the end, the best output interface is selected at this stage for packet forwarding (I-Pkt).

3-2-11-At this stage, if the time of simulation is not over, it returns to step 1 in the flowchart and these stages are repeated; otherwise, it ends.

It is assumed that the round trip of the packet is in one direction but inversely. If in the path forward, an optimal path is found for a packet, the packet returns in the same path but inversely to reach the user.

4. SIMULATION RESULTS

4.1. Simulation Environment and Tools:

ndnSIM[3] software and the NDNs simulator have been used to do simulation tests. ndnSIM software [19] has been implemented based on NS3[4] software framework [20, 21] (simulator of computer networks and the upgraded version of NS2 software [22]) in which all modules of NS3 can be run in this software. ndnSIM [23] software has the following specifications:

4-1-1-Using common protocols: protocols like P2P (Peer-to-Peer) was implemented in this software and it can easily be used in different scenarios and examples by calling its library functions.

4-1-2-Using graphic environment: tracking the happenings in a graphic environment using a module or capability called visualizer.

4-1-3-Development capability: as this simulator was written in two languages of C++ and Python and using the concept of object-oriented in this software, the flexibility of the software is very high and having object-oriented and programming knowledge, we can easily write our library and call it.

4-1-4-The following parameters were considered to assess and compare the methods.

4-1-5-Cost: it is the number of request packets which came out of the router to reach the other router [7]. Here the mean of the sum of output request packets from the routers divided by the number of routers is considered to be equal the cost parameter.

4-1-6-Delay: the better the operation of finding an optimal path from the consumer to producer, it means the delay in packet forwarding (I-Pkt) until packet receiving (D-Pkt) is lower and this means that a more optimal path comparing the previous methods was found. Here the delay parameter is equal the sum of the packets that reached the destination in the real time that is subtracted from the time that the packet reaches the destination in the equation and their mean will be computed.

4-1-7-Hit Ratio: the mean of responded packets in each router proportional to the total number of passing request packets from the routers for all network routers.

4-1-8-Hop count: the number of steps equals the mean of passed edges by request packets to reach the destination. The better the performance in two aforementioned parameters, the lower will be the number of steps that a packet must pass the routers to reach its D-Pkt and this means the reduction of network traffic and congestion.

For simulation and its related parameters, the base article has been followed in a way that the number of nodes was considered as 64 and in a 20 second simulation time with three amounts of packet request by the consumer and the mentioned Zipf's law coefficient, the amounts of this simulation output have been examined in Table 3: simulation model and parameters. Each router is attached to one repository.

[3] Named data networking SIMulator

[4] Network Simulator v3

We also used 10000 contents with three replicas randomly allocated to different repositories.

Table 3. Simulation parameters.

Parameter	Values	Description
Topology type	Abilene Topology	The used topology based on the basic article consisted of determined number of users and producers and is an 8*8 topology.
Number of nodes	64	20 consumers, 12 producers and 32 routers
Number of I-Pkts	10, 20, 30	Number of I-Pkts in every 20 seconds
Zipf's law Coefficient	0.8, 1, 1.2	This simulation is done with one of Zipf's law parameters in each simulation period.
Traffic type	Zipf's law	Constant bit rate and Zipf's law traffic are done. Here Zipf's law has been used.

4.2. Mean Cost Evaluation

As observed in Table 3, Zipf's law coefficient for the number of request packets by the user is the three values of 10, 20 and 30. Fig. 3 shows the Mean Cost in ACO and the IWO. The Mean Cost in both ant colony optimization algorithm and the proposed method increases by I-Pkt number increase forwarded by the consumers. Having velocity parameter with constant value has caused the increase of packet hit ratio in the router's caching and decrease of packets coming out of the router comparing ant colony optimization method.

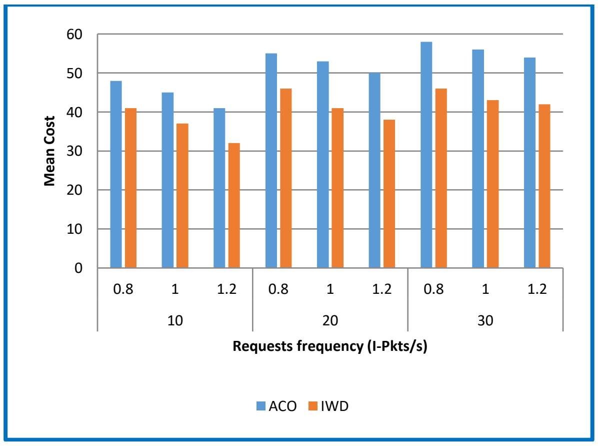

Fig. 3. Mean cost comparison between ACO and the proposed method.

4.3. Mean Delay Evaluation

In Fig. 4, I-Pkt mean delay in reaching the producer has been shown in ant colony optimization algorithm and the proposed method. As shown in this Figure, reducing the network congestion and traffic through cost parameter optimization in the previous part, we can decrease I-Pkt delay in reaching the producer. Increasing the I-Pkt number does not change this parameter significantly comparing ant colony optimization algorithm.

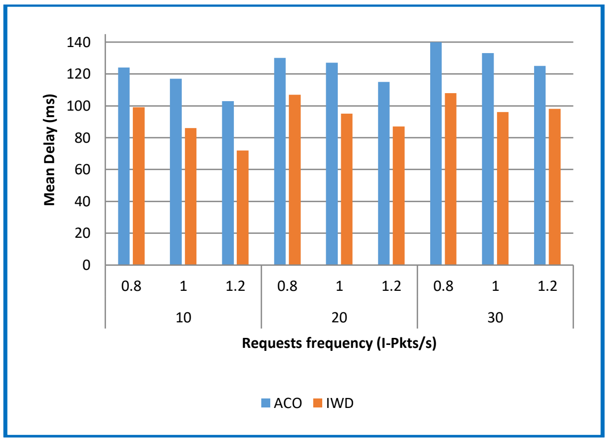

Fig.4. Mean Delay Comparison between ACO and the proposed method.

4.4. Mean Hit Ratio Evaluation

As observed in Fig. 5, the mean hit ratio in ant colony optimization algorithm and the proposed method have been shown. By increasing the number of request packets, hit ratio in both methods will increase; however, as the proposed method performs better in saving data packets in the routers' caching regarding the given time, it causes the request packets in the routers reach the D-Pkt faster through searching the caching and this means hit ratio increase.

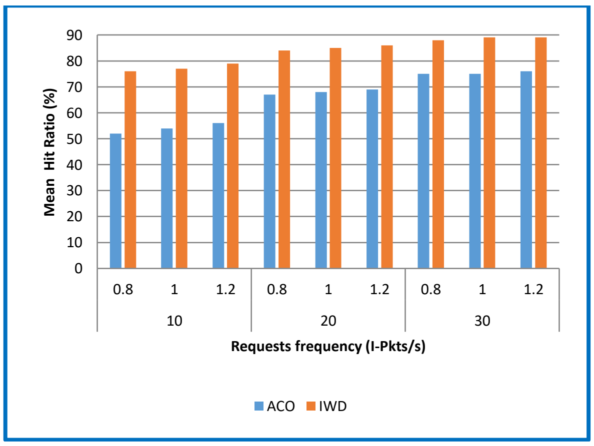

Fig. 5. Hit ratio mean in receiving the D-Pkt by the I-Pkt in the router's caching between ACO and the proposed method.

4.5. Mean Hop Count Evaluation

As shown in Fig. 6, by improving the quality parameters in the previous parts which were related to each other, they in turn performed better than ant colony optimization algorithm and caused the I-Pkt to reach the given D-Pkt after lower number of hops. The factor which causes the proposed method perform better than ant colony optimization algorithm is its structure which causes the D-Pkt to be saved in the router's caching in the return path and therefore, there will be no need to pass many routers by the I-Pkt to reach its D-Pkt.

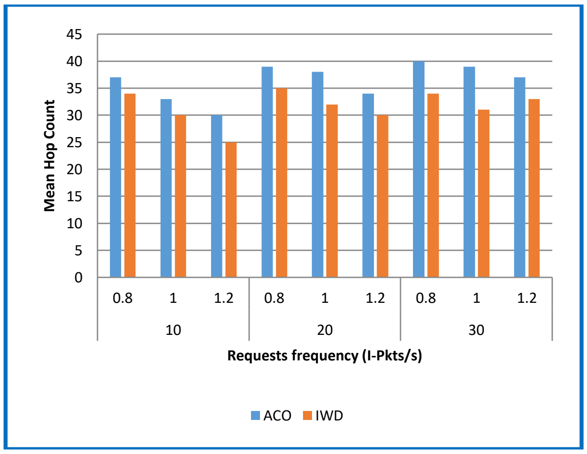

Fig. 6. The mean of number of steps in reaching the I-Pkt from the source to the destination.

In Table 4, the percentages of the proposed method improvement comparing ant colony optimization algorithm have been shown for the parameters of cost, delay, hit ratio and hop count.

Table 4. Percentages Improvement by the proposed method in Comparison with ACO.

Zipf's law coefficient	I-Pkt Rate			Cost	Delay	Hit Ratio	Hop Count
	10	20	30				
0.8	*			15%	21%	32%	7%
1	*			18%	27%	30%	12%
1.2	*			22%	31%	30%	18%
0.8		*		17%	18%	31%	10%
1		*		23%	26%	20%	17%
1.2		*		24%	25%	20%	12%
0.8			*	21%	23%	15%	15%
1			*	24%	28%	16%	21%
1.2			*	23%	22%	15%	10%

5. CONCLUSION AND FUTURE WORK

Each network needs a packet forwarding method for information transfer. Different methods are used for packet forwarding in an optimized way. In this study, intelligent water drop algorithm has been used. In simulation part, the results of the proposed method of this study have been compared in different scenarios and in terms of different assessment criteria (cost, delay, hit ratio and hop count) with ant colony optimization algorithm. Furthermore, the service quality parameters have also been used in affecting the results output as the optimal path in a network that is not only the shortest one and cases like network congestion and traffic must also be considered. According to the tables, the results show that the considered parameters in the simulation have improved under the effect of service quality parameters comparing ant colony optimization algorithm. Finding an optimal path in a network which depends on significant service quality parameters can be used and examined in the next implementations and simulations. As an example, one of service quality parameters which has not been examined in the simulation is the mean of request packet waiting time in the router to determine the next path to forward to the next router. For future research, the following guidelines are suggested for improving the proposed method:

- Predicting future internet interest requests for more accurate routing and storing the data in the return path,
- Combining the proposed method with other meta-heuristic algorithms to optimize the coefficients of fitness function,
- Combining the proposed method with cache storage methods to improve routing.

Finally, stationary routers in a wired network are considered in this study. As a weakness point, it is better to consider mobile routers in a wireless environment to measure the efficiently of the proposed algorithm more precisely and completely. Moreover, it is possible to improve the effectiveness of the selection phase in IWD algorithm using linear or exponential ranking.

REFERENCES

[1] D. Saxena, V. Raychoudhury, N. Suri, C. Becker, and J. Cao, **"Named Data Networking: A survey,"** *Comput. Sci. Rev.*, Vol. 19, pp. 15–55, 2016.

[2] L. Zhang *et al.*, **"Named Data Networking,"** *SIGCOMM Comput. Commun. Rev.*, Vol. 44, No. 3, pp. 66–73, Jul. 2014.

[3] E. Lee, **"Categories and Subject Descriptors,"** *Proc. 47th Des. Autom. Conf. (DAC),ACM*, pp. 737–742, 2010.

[4] Z. Rezaeifar, J. Wang, H. Oh, S.-B. Lee, and J. Hur, **"A reliable adaptive forwarding approach in named data networking,"** *Futur. Gener. Comput. Syst.*, Vol. 96, pp. 538–551, 2019.

[5] A. Kalghoum and L. A. Saidane, "**FCR-NS: a novel caching and forwarding strategy for Named Data Networking based on Software Defined Networking,"** *Cluster Comput.*, Vol. 22, No. 3, pp. 981–994, 2019.

[6] M. Zhang, X. Wang, T. Liu, J. Zhu, and Q. Wu, **"AFSndn: A novel adaptive forwarding strategy in named data networking based on Q-learning,"** *Peer-to-Peer Netw. Appl.*, Vol. 13, No. 4, pp. 1176–1184, 2020.

[7] C. Li, K. Okamura, and W. Liu, **"Ant colony based forwarding method for content-centric networking,"** *Proc. - 27th Int. Conf. Adv. Inf. Netw. Appl. Work. WAINA 2013*, pp. 306–311, 2013.

[8] Q. Huang and F. Luo, "**Ant-colony optimization**

based QoS routing in named data networking," *J. Comput. Methods Sci. Eng.*, Vol. 16, No. 3, pp. 671–682, 2016.

[9] A. Kerrouche, M. R. Senouci, A. Mellouk, and T. Abreu, **"AC-QoS-FS: Ant colony based QoS-aware forwarding strategy for routing in Named Data Networking,"** *2017 IEEE Int. Conf. Commun.*, pp. 1–6, 2017.

[10] A. Kerrouche, M. R. Senouci, and A. Mellouk, **"QoS-FS: A new forwarding strategy with QoS for routing in Named Data Networking,"** *2016 IEEE Int. Conf. Commun. ICC 2016*, 2016.

[11] X. Zeng and Z. H. Gao, **"Rank-based routing strategy for named data network,"** *Appl. Mech. Mater.*, Vol. 543–547, pp. 3320–3323, 2014.

[12] C. Yi, A. Afanasyev, I. Moiseenko, L. Wang, B. Zhang, and L. Zhang, **"A case for stateful forwarding plane,"** *Comput. Commun.*, Vol. 36, No. 7, pp. 779–791, 2013.

[13] M. Kuai, X. Hong, and Q. Yu, **"Delay-tolerant forwarding strategy for named data networking in vehicular environment,"** *Int. J. Ad Hoc Ubiquitous Comput.*, Vol. 31, No. 1, pp. 1–12, 2019.

[14] Y. Tao, Y. Fan, and R. Huang, **"A probability forwarding strategy for Named Data Networking based on congestion level of interface,"** *Int. J. Smart Eng.*, Vol. 3, pp. 1–6, 2019.

[15] Z. Qureshi *et al.*, **"Fuzzy Logic-based Efficient Interest Forwarding (FLEIF) in Named Data Networking,"** *Trans. Emerg. Telecommun. Technol.*, Vol. 30, No. 9, p. e3577, 2019.

[16] T. Cheng, Y. Yang, and D. Qu, **"A Forwarding Strategy Based on Recommendation Algorithm in Named Data Networking,"** in *2019 IEEE 21st International Conference on High Performance Computing and Communications*, 2019, pp. 51–53.

[17] M. Abdelaal, M. Karadeniz, F. Durr, and K. Rothermel, **"LiteNDN: QoS-Aware Packet Forwarding and Caching for Named Data Networks,"** *2020 IEEE 17th Annu. Consum. Commun. Netw. Conf. CCNC 2020*, January, pp. 10–13, 2020.

[18] S. Mastorakis, A. Afanasyev, and L. Zhang, **"Public Review for On the Evolution of ndnSIM Artifacts Review for On the Evolution of ndnSIM,"** *ACM SIGCOMM Comput. Commun. Rev.*, Vol. 47, No. 3, pp. 19--33, 2017.

[19] A. Afanasyev, I. Moiseenko, and L. Zhang, **"ndnSIM: NDN simulator for NS-3,"** *NDN, Tech. Rep. NDN-0005*, pp. 1–7, 2012.

[20] S. Mastorakis, A. Afanasyev, I. Moiseenko, and L. Zhang, "**ndnSIM 2 : An updated NDN simulator for NS-3,"** *Dept. Comput. Sci., Univ. California, Los Angeles, Los Angeles, CA, USA, Tech. Rep. NDN-0028*, pp. 1–8, 2016.