NDN congestion control based on fuzzy comprehensive evaluation algorithm

Gaolin Cui*, Hui Wang2, Zhihui Fan3, Peiyu Li4

1School of Information Engineering, Henan University of Science and Technology, Zhumadian, Henan, 471000, China
2School of Information Engineering, Henan University of Science and Technology, LuoYang, Henan, 471000, China
3School of Information Engineering, Henan University of Science and Technology, LuoYang, Henan, 471000, China
4School of Information Engineering, Henan University of Science and Technology, LuoYang, Henan, 471000, China

* Corresponding author’s e-mail: 180318050257@stu.haust.edu.cn

Abstract. The congestion control algorithm of the named data network is not accurate in obtaining the congestion status, and the research is not carried out in time, and a congestion control strategy based on the fuzzy comprehensive evaluation algorithm is proposed. This strategy uses the fuzzy comprehensive evaluation algorithm to calculate and analyze the above two network parameters to obtain the network congestion state by detecting the buffer queue length of the network intermediate node and the actual and predicted round-trip time ratio. In order to refine the degree of network congestion, a variety of congestion is designed. The feedback signal uses the existing NACK feedback mechanism of the NDN network to send a congestion signal to the downstream node, and the downstream node adaptively adjusts the interest packet sending rate according to the congestion signal to achieve the purpose of reducing network congestion. In the NDN-SIM simulator, this algorithm is compared and analyzed with mainstream algorithms, and it is confirmed that this strategy can make full use of network resources and has higher transmission efficiency.

1.Introduction

The NDN network is a content-centric future network architecture. It overcomes the disadvantages of the TCP/IP protocol in principle, and changes the communication mode from host address-oriented content-oriented, focusing only on the "what" of the content. It does not care about the content "where" [1], and is especially suitable for Internet scenarios where the communication object is moving. In order to ensure the stability and quality of service of the NDN network, it is necessary to study the NDN network congestion control [2]. The architecture of the NDN network is based on the receiver-driven and has the characteristics of multi-source and multi-path [3], and the TCP/IP network cannot simply be used to determine whether the network is congested based on the delay. However, the NDN network supports traffic balance, that is, an interest packet return at most one data packet. According to this feature, some scholars have proposed to adjust the interest packet sending rate by judging whether the network link is congested based on the length of the interest packet buffer queue. When the queue length is small, increase the interest packet sending rate, when the buffer queue length
is larger, reduce the interest packet sending rate, and the network environment is uncontrollable, In the case of relatively good network performance, this method can improve network throughput, but when the network performance is not good and the length of the cache queue is small, this method will aggravate network congestion. This kind of complex network is difficult to accurately judge the state of the network based on a certain parameter, and the fuzzy comprehensive evaluation algorithm is a comprehensive evaluation method based on fuzzy mathematics, which makes an overall evaluation of things restricted by multiple factors, so this article uses The fuzzy comprehensive evaluation algorithm judges the network status by obtaining multiple network parameters.

This paper proposes NDN congestion control based on fuzzy evaluation algorithm (Congestion Control Based on Fuzzy Evaluation, FE-CC). The specific implementation process is to design four network congestion states: network idle, network normal, network busy, network congestion; then obtain two network parameters: the average queue length of interest packets and the round-trip time ratio; Then use the fuzzy comprehensive evaluation algorithm to calculate the network parameters to get the network status, the routing node feeds back the network state signal to the downstream node, and the downstream node adjusts the sending rate of the interest packet according to the network state signal to improve the overall performance of the network.

2. Congestion control algorithm

2.1 Fuzzy comprehensive evaluation model

Fuzzy comprehensive evaluation is fuzzy multi-objective decision-making. It is a method of making comprehensive decisions on affairs by using fuzzy transformation for a certain purpose in a fuzzy environment, considering the influence of multiple factors.

Establish factor set U and state set V according to Zadeh’s definition of fuzzy subset: select the Average Queue Length (Average Queue Length, AQL) and Round Trip Time Ratio (Round Trip Time Ratio, RTTR) as the factor set parameters for fuzzy comprehensive evaluation, and state The set parameters are NACK_FREE, NACK_NORMAL, NACK_BUSY, NACK_CONGESTION. That is: U=(u1,u2): u1 is AQL, u2 is RTTR; V=(v1,v2,v3,v4): v1 is NACK_FREE, v2 is NACK_NORMAL, v3 is NACK_BUSY, v4 is NACK_CONGESTION. Take 4 mappings from U to the closed interval [0,1], determine the four fuzzy subsets of U: F, N, C, M, μF(u), μN(u), μC(u), μM(u) correspond to the membership functions of F, N, C, and M, respectively. At the same time, the decision set V=(v1, v2, v3, v4) is used to describe the current congestion state of the network.

In order to obtain an accurate current network congestion status, fuzzy comprehensive evaluation is used to comprehensively evaluate the state of the NDN network. There is a fuzzy relationship from U to decision set V, which is represented by a fuzzy matrix R, denoted as:

\[ R = \begin{bmatrix} \mu_F(u_1) & \mu_N(u_1) & \mu_C(u_1) & \mu_M(u_1) \\ \mu_F(u_2) & \mu_N(u_2) & \mu_C(u_2) & \mu_M(u_2) \end{bmatrix} \]  

(1)

Each factor in the factor set has a different impact on the state set transaction, so each factor is assigned a corresponding weight, which is a fuzzy vector in the universe of discourse, denoted as:

\[ W = (\omega_1, \omega_2) \in F(U) \]  

(2)

Where \( \omega_1 \) is the weight of the first factor in U, and \( \omega_2 \) is the weight of the second factor in U, and satisfies

\[ \omega_1 + \omega_2 = 1 \]  

(3)

The fuzzy comprehensive evaluation results are:

\[ B = W \cdot R = (b_1, b_2, b_3, b_4) \]  

(4)

Among them: B is the fuzzy subset on the decision evaluation set V, bj is the degree of membership of the level Vj to the comprehensive evaluation fuzzy subset B. Comparing bj \( j=1,2,3,4 \), there is a k such that

\[ B_k = \max \{b(j)\} \]  

(5)

According to the principle of maximum membership degree, the network is in the state determined
by $b_k$, and the solution is selected.

2.1.1 Selection of factor set

(1) Average buffer queue length of interest packets

The internal mechanism of the NDN network is that one interest packet corresponds to one data packet, so the interest packet buffer queue is used to determine how many data packets will exist in the current network link, and the network status is judged according to the network link capacity. In order to avoid the impact of burst traffic on the judgment of the network status, this paper divides a measurement period $T$ into $i$ segments evenly, and monitors the length of the interest packet buffer queue every $T/i$ time. A total of $i+1$ times are monitored in a period, and these measured values is the average interest packet queue length $AQL$ for a period. The calculation formula of $AQL$ is:

$$AQL = \frac{\sum_{j=1}^{i+1} \text{QueueLength}_{\text{current}}}{i+1}$$

Where $\text{QueueLength}_{\text{current}}$ represents the length of the interest packet buffer queue of the current port.

In order to reflect the changing trend of network congestion, $AQL$ is optimized. Each monitored interest packet queue length corresponds to a weight value, and the weight value increases monotonically within a period. Assuming that $\text{QueueLength}_{\text{current}(m)}$ and $\text{QueueLength}_{\text{current}(k)}$ are the instantaneous interest packet queue length monitored in a cycle ($m < k$), the weight $E_m$ of $\text{QueueLength}_{\text{current}(m)}$ is less than the weight $E_k$ of $\text{QueueLength}_{\text{current}(k)}$. The use of this weight-increasing design shows that the length of the newly detected interest packet queue is more important than the historical queue length, and it can better reflect the current network congestion. In order to enable the weight to grow smoothly, the algorithm adopts a linear growth method, and the weight calculation formula is as follows:

After optimization, the final average interest packet queue length $AQL_{\text{final}}$ is calculated as follows:

$$AQL_{\text{final}} = \frac{\sum_{j=1}^{i+1} E_j \cdot \text{QueueLength}_{\text{current}(j)} }{i+1}$$

(2) Round-trip time ratio $RTTR$

The round-trip delay reflects the state of the network to a certain extent. This article uses some processing on the round-trip delay as a factor of the fuzzy comprehensive evaluation algorithm.

the calculation formula of $RTO$ as:

$$RTO_i = \beta RTT_{\text{previous}} + (1 - \beta) RTT_i$$

$\bar{RTT}$ is the average value of the previous round-trip time, $RTT_i$ is the farthest measured round-trip time of the most recently received data packet, and $\beta$ is a constant from 0 to 1. The timeout retransmission time is as follows:

$$RTO_i = \bar{RTT} + f \cdot \sigma$$

$f$ is to adapt to the constant change of $RTT$ to prevent frequent premature timeouts, and $\sigma$ is the estimated value of $RTT$ standard deviation.

The $RTO$ calculated here is stored in the data requesting node as the time for the interest packet to request timeout retransmission. What this article has done is to store a time in the intermediate node as a condition for network congestion judgment. Therefore, intermediate nodes need to recalculate $RTO$.

$$\text{Data}_{\text{size}} = \frac{\text{package}_{\text{DataSize}}}{\text{package}_{\text{InterestSize}}} \cdot \text{Interest}_{\text{size}}$$
Since an interest packet corresponds to a data packet and the data packet is returned in the original way, the data is transmitted as a bit stream on the network link, so the transmission time ratio of the data packet and the interest packet can be the ratio of the size of the data packet to the interest packet calculation.

The transmission time of the interest packet from the sender to the intermediate node is obtained as Interesttime through the timestamp. Since the return of the data packet is a special mechanism for returning according to the requested route, the time of the data packet from the intermediate node to the requesting end is Datatime, and the intermediate node can determine the timeout time INT of the content node that satisfies the request is RTO-(Interesttime+Datatime), and the intermediate node maintains this time value. When the intermediate node forwards the interest packet, re-timing, calculate the actual time RT when the intermediate node receives the data packet sent back by the content node, compare the ratio of RT and INT to determine which state the network is in, and take the corresponding solution method.

\[ RTTR = \frac{RT}{INT} \]  

2.1.2 Membership function

The choice of membership function is the key to fuzzy comprehensive evaluation and analysis. The commonly used membership functions include trapezoidal function, trigonometric function, Gaussian function, and so on. Each factor set has different membership functions corresponding to different congestion levels. After reasoning and analysis, this article adopts the combination of trapezoidal function and Gaussian function as the membership function on the universe U. As shown in Figure 1:

The parameters of the abscissa have different meanings under different factors. Under the u_1 factor, it represents the average queue length. AQL_{final} is calculated according to formula (9), and the area where the AQL_{final} value is in the abscissa is determined, and then the corresponding membership function is used to calculate the degree of membership of the network state. Under the u_2 factor, these parameters represent the ratio of the predicted round-trip time to the actual round-trip time. Calculate RTTR according to formula (13), determine that the RTTR value is in the area where the abscissa is located, and then use the corresponding membership function to calculate the degree of membership of the network state.

Determine the fuzzy matrix of the network parameters to the network state according to the calculated membership degree, and then use the formulas (4) and (5) to determine the network state.

2.2 Setting of NACK packet

FE-CC aims to improve network throughput, expand NACK packets, and add four new NACK packets:

1) NACK_FREE: Determine whether the network is idle
2) NACK_NORMAL: Determine whether the network is normal
3) NACK_BUSY: Determine whether the network is busy
4) NACK_CONGESTION: Determine whether the network is congested
This algorithm has a small granularity for dividing the network state and can adapt to a complex network environment. The network status is monitored at regular intervals, and the corresponding NACK packet is fed back to the downstream node. The reason for the design of the NACK_FREE package is to adjust the sending rate to occupy the bandwidth quickly when the network is judged to be idle. The design of the NACK_NORMAL package is to prevent the impact of burst traffic on network performance.

2.3 Rate adjustment algorithm
The rate adjustment algorithm is based on the AIMD algorithm and uses a display feedback mechanism. The routing node learns the specific network congestion status to adjust the interest packet sending rate. When the NACK_FREE packet is received, the network is in an idle state. In order to occupy the idle resources of the network as soon as possible, the rate adopts the multiplicative increase (MI) algorithm; when the NACK_NORMAL packet is received, the rate adopts an additive increase (Additive Increase). Increase, AI) algorithm can prevent the impact of burst traffic on the network; when the NACK_BUSY packet is received, it indicates that the network is in a saturated state and the sending rate should be quickly reduced. Therefore, the Multiplicative Decrease (MI) algorithm is adopted.; When the NACK_CONGESTION packet is received, the network is already in a high load state. At this time, other suitable ports are selected for the request packet to slow down the load of the current link, so that the network can run stably. The specific algorithm is as follows:

\[ \text{MI: } \text{SendRate}(t+ep) = \text{SendRate}(t) \ast (1 + \zeta) \quad (14) \]
\[ \text{AI: } \text{SendRate}(t+ep) = \text{SendRate}(t) + \vartheta \quad (15) \]
\[ \text{MD: } \text{SendRate}(t+ep) = \text{SendRate}(t) \ast \psi \quad (16) \]

Among them: ep is an evaluation period, \( \zeta, \vartheta, \psi \) are adjustment parameters.

3. Simulation analysis of algorithm performance
This paper implements the proposed FE-CC algorithm in the simulation platform NDN-SIM, and uses different topologies to evaluate the algorithm. In order to clearly see the impact of the proposed algorithm on the network performance, the widely recognized HR is selected. -ICP[4] and ARMN[5] are used as benchmark algorithms to compare and analyze the three evaluation indicators of packet loss rate, average delay and throughput.

The basic settings of the experiment are as follows:

a) The experimental topology uses dumbbell and multi-path topology as shown in Figure 3 and Figure 4. Consumers request data at a rate of 500 Interest packets per second, and stop the request after 10 seconds

b) The link bandwidth between the content requester and the intermediate node is set to 10Mbps, and the link bandwidth between the intermediate nodes is set to 2Mbps

c) The size of the interest packet is fixed at 32Byte, and the size of the data packet is fixed at 1024Byte

d) Under the condition of the average cache queue length represented by factor 1 in Figure 1, take

\[ \alpha = \frac{1}{10}, \quad \beta = \frac{1}{2}, \quad \gamma = \frac{3}{5}, \quad \eta = \frac{4}{5}, \quad \delta = \frac{1}{1} \]

e) Take under the condition of the ratio of round-trip time represented by factor 2 in Figure 1

\[ \alpha = \frac{1}{5}, \quad \beta = \frac{1}{2}, \quad \gamma = \frac{7}{10}, \quad \eta = \frac{1}{1}, \quad \delta = \frac{7}{5} \]
Figures 4 and 5 show how the average delay varies with the number of interest requests. When the number of interest requests is greater than 1250, the three algorithms gradually show different performance. The algorithm proposed in this paper predicts the state of the network at the intermediate node, sets up a variety of congestion signal feedback mechanisms through the fuzzy evaluation algorithm, and quickly adjusts the interest packet sending rate. When the network is in congestion, intermediate nodes can also choose other available ports independently, so the lowest average delay is always maintained.

In ARMN, the end node uses NACK packets as congestion information to adjust the window, but because the network is in congestion, the return of the NACK packet will be delayed and will further increase network congestion, causing the end node to be unable to obtain the network status in time. Circumstances, causing the delay to become larger. The HR-ICP end node sets a window adjustment strategy based on the round-trip delay, which can control the round-trip delay at the end node to a
certain extent.

(2) Packet loss rate

Figures 6 and 7 show how the packet loss rate varies with the number of requests. In the two topologies, the trend of the packet loss rate with the number of interest requests is roughly similar. When the number of interest requests is small, the packet loss rate varies with the number of requests. The increase in the number of interest requests is relatively stable; when the number of interest requests gradually increases, the three congestion control algorithms gradually exhibit different performance. Among them, the algorithm in this paper sets a rate adjustment mechanism based on multiple congestion signals at the node, and the design of the congestion feedback signal is more able to meet the actual state of the network, and a timeout timing mechanism based on the NDN multi-source network scenario is set at the end node, so packet loss The lowest rate. HR-ICP uses RTO as a congestion signal. When a packet timeout occurs, it is considered as congested and the congestion window is multiplicatively reduced to control the packet loss rate to a certain extent; ARMN uses the NACK packet as a congestion signal to adjust the window at the end node. The NACK packet needs to reach the end node through hop-by-hop backtracking. The end node cannot obtain the network status in time to adjust the rate, resulting in the largest packet loss rate.

![Figure 6](image6.png)  
**Figure 6** The change of the packet loss rate with the number of interest requests in the dumbbell topology

![Figure 7](image7.png)  
**Figure 7** The change of packet loss rate with the number of interest requests in a multipath topology

(3) Throughput

Figures 8 and 9 show how the throughput varies with the sending time of the interest packet. When the number of interest requests is small, the trend of throughput increasing with the increase of the number of interest requests is more obvious; when the number of interest requests is greater than 2000, the throughput is no longer affected by the number of interest requests, and different mechanisms
gradually reflect different performance. The ICP adjustment mechanism adopted by HR-ICP at its end nodes is mainly based on RTT. When the RTT is too large, the window will be drastically reduced, resulting in a small port queue, and the throughput of the most terminal node is also relatively low. ARMN Utilizing the idea of multi-path forwarding, when the network is congested, the NACK packet is re-searched for the interface instead of delayed forwarding, which ensures the throughput. On the one hand, the algorithm in this paper monitors the network status at the intermediate node and adjusts the sending rate adaptively. The MIAIMD algorithm is used to adjust the sending rate, which quickly occupies the link bandwidth when the network is idle. On the other hand, the node rate adjustment algorithm cannot adapt well. The network will look for other available interfaces again, so the FE-CC algorithm can make full use of network resources and always maintain a high throughput.

![Fig. 8 The throughput of the dumbbell topology varies with the sending time of the interest packet](image1)

![Fig. 9 The throughput in the multipath topology changes with the sending time of the interest packet](image2)

4. Conclusion

Aiming at the problem of inaccurate and untimely acquisition of congestion status by NDN's own congestion control algorithm, this paper analyzes the existing congestion control algorithms of NDN and proposes a congestion control based on a fuzzy comprehensive evaluation algorithm. The algorithm uses the buffer queue length of the intermediate node and the ratio of actual to predicted round-trip time, and uses a fuzzy comprehensive evaluation algorithm to calculate and analyze the above two network parameters to obtain the network congestion state. In order to accurately obtain the degree of network congestion, a variety of congestion feedback signals are designed, and the intermediate node adjusts the rate according to different congestion signals.

Experiments show that the algorithm has a significant improvement compared with mainstream algorithms in terms of average delay, packet loss rate, and throughput. In the next study, the impact of caching strategies for congestion control will be analyzed, and the forwarding, caching, and
congestion control of interest packets by NDN intermediate nodes will be comprehensively considered to make congestion control more effective in NDN networks.

Acknowledgements
This article is one of the phased achievements of the key scientific research project (18A520007) of higher education institutions in Henan Province and Research on Heterogeneous Network Congestion Control Based on IPV6 (NGII20180107).

References
[1] Bourtsoulatze, E., Member, IEEE, Thomos, N., & Member, S.. (2015). Content-aware delivery of scalable video in network coding enabled named data networks. *IEEE Transactions on Multimedia, PP*(99), 1-1.
[2] Yu, Y., Li, Y., Du, X., Chen, R., & Yang, B.. (2018). Content protection in named data networking: challenges and potential solutions. *Communications Magazine, IEEE, 56*(11), 82-87.
[3] Ren, Y., Li, J., Shi, S., Li, L., Wang, G., & Zhang, B.. (2016). Congestion control in named data networking – a survey. *Computer Communications, 86(JUL.15)*, 1-11.
[4] Mastorakis, S., Afanasyev, A., & Zhang, L.. (2017). On the evolution of ndnsim: an open-source simulator for ndn experimentation. *ACM SIGCOMM Computer Communication Review, 47*(3), 19-33.
[5] Lee, S., Kim, Y., Yeom, I., & Kim, Y.. (2017). Active request management in stateful forwarding networks. *Journal of Network & Computer Applications, 93*(sep.), 137-149.