International Journal of Computer Science and Information Technology
ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 3, Number 1, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v3n1.09
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Review of Congestion Control Methods in Named Data Network

Bohan Zhang

Henan University of Science and Technology, Luoyang 471000, China

ABSTRACT

In recent years, with the development of new networking technologies, the demand for networks has shifted from simple resource sharing to large-scale content distribution. This change has led to the gradual inadequacy of the traditional TCP/IP architecture in meeting the current needs of production and daily life. In this context, Named Data Networking (NDN) has emerged. NDN networks, with their content-centric nature and features such as in-network caching, multi-source forwarding, and routing control, have attracted significant attention from researchers both domestically and internationally. However, the novel architecture of the network also presents new challenges for congestion control research in NDN. This paper conducts thorough research on the related work of congestion control in NDN, provides a detailed overview of existing research efforts in NDN congestion control, and describes potential future research directions in NDN congestion control. The aim is to provide valuable references for future research on congestion control in NDN.

KEYWORDS

Named data networking; Congestion control; Receiver-driven

1. INTRODUCTION

In recent years, the functionality of the Internet has shifted from resource sharing to large-scale content distribution and retrieval. Traditional network architectures have gradually become inadequate to meet this transformation [1]. In order to provide better network services and meet user needs, researchers both domestically and internationally have collaborated to develop Named Data Networking (NDN) [2]-[4].

NDN is a network architecture that centers around named data, allowing information retrieval and routing based on names. The name-centric architecture of NDN decouples content from its location, providing greater flexibility. Additionally, NDN natively supports features such as multi-source communication, multipath routing, and network caching.

In addition to the new network architecture, congestion control mechanisms play a crucial role in network transmission. Congestion control mechanisms not only directly determine the actual performance of the network but also have a significant impact on users' network experience. As shown in Figures 1, in an ideal scenario, a comprehensive congestion control mechanism can fully utilize network resources and ensure a satisfactory user experience for all users. However, in real-life situations, the actual usage of the network can vary significantly due to factors such as link load, router load, and user scale. Moreover, the impact of congestion on network operation varies under different network loads, ranging from busy to congested and from congested to deadlock. Each instance of congestion intensification can lead to more severe issues, such as packet loss, link instability, and increased access time. Therefore, research on congestion control is of utmost importance in addressing network congestion issues.

However, these rich mechanisms also make congestion control in NDN more complex. This paper focuses on congestion control in NDN, starting from the perspective of network congestion. The organization of this paper is as follows: Section 1 serves as the introduction, Section 2 introduces the relevant content of network congestion, Section 3 presents the current research on congestion control in NDN, including receiver-driven congestion control, hop-by-hop congestion control mechanisms, hybrid congestion control mechanisms, and other congestion control mechanisms in NDN. Section 4 will discuss future research directions for congestion control in NDN.

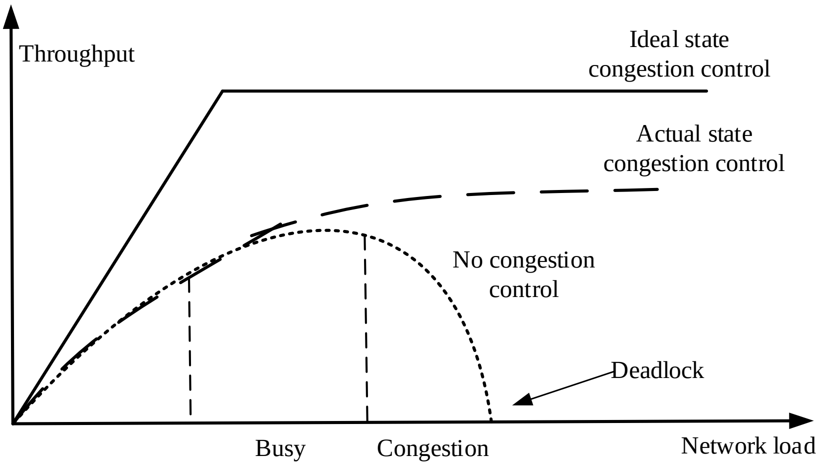
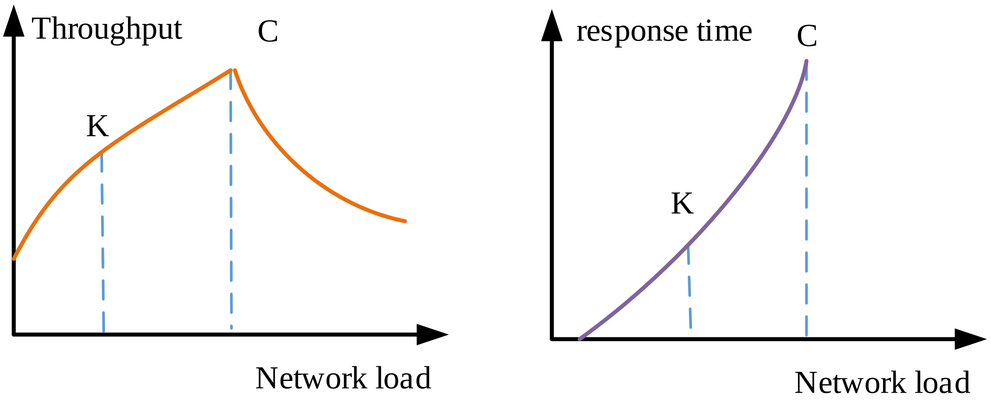


Figure 1. Congestion Control Mechanism Function

2. INFORMATION RELATED TO NETWORK CONGESTION

Network congestion is a common phenomenon in which the actual amount of data transmitted during a specific period exceeds the capacity of the network link, resulting in a degradation of network performance and impacting user experience [5].



(a) (b)

Figure 2. Throughput response time vary with network load

Figure 2 depicts the relationship between throughput and response time as network load varies [6]. In the early stages of network operation, as user requests increase, the amount of data that needs to be transmitted through the network also increases, leading to linear growth in throughput, as shown in Figure 2(a). Along with this, the response time gradually increases, as indicated by point K in Figure 2(b). After reaching point K, the throughput continues to increase as the network load approaches its peak. However, once it exceeds the throughput threshold at point C, the network becomes overloaded, leading to severe packet loss, which is a significant cause of the sudden drop in throughput. At this stage, the response time also reaches its maximum, resulting in a poor user experience, as shown in Figure 2(b). Through analysis, it can be inferred that the user experience is better when the network load is around point K, while it becomes extremely poor beyond point C. Therefore, congestion control aims to stabilize the utilization of network resources near point K.

Therefore, being able to dynamically adjust resource utilization according to needs and coordinate resource allocation across different parts of network transmission is what holds greater practical value.

3. CURRENT RESEARCH ON NDN CONGESTION CONTROL

3.1. Receiver-Driven Congestion Control

Receiver-Driven Congestion Control mechanism is similar to traditional control methods, where the data consumer side restricts the rate of interest packet transmission to achieve congestion control objectives.

ICP [7] is a typical example of a receiver-driven algorithm. In ICP, the receiver maintains an interest packet expiration timer on its side. If the timer expires, it is considered as a packet loss. To achieve reliable data transmission, the consumer retransmits the interest packet after a packet loss. By employing this approach, the consumer determines congestion control not based on packet loss but instead relies on the timer, as shown in Equation 1.

Where RTTmin and RTTmax are the calculated from the history of receiving consecutive 20 data packets (excluding retransmitted packets). Additionally, ICP also utilizes AIMD as a congestion window adjustment mechanism, allowing it to quickly acquire network bandwidth resources. Moreover, to address resource allocation issues between different interest flows, ICP incorporates a bandwidth allocation mechanism. This mechanism enables different interest flows to share routing resources and obtain equal downstream bandwidth, facilitating efficient resource utilization and fair distribution.

$$\tau = RTT_{\min} + (RTT_{\max} \text{-} RTT_{\min}) \times \delta \tag{1}$$

To make ICP suitable for multi-path scenarios, the RAAQM protocol [8] was proposed. In RAAQM, each data packet adds router identifiers during forwarding. Then, the receiver detects the RTO (Retransmission Time Out) for each route based on different sequences of router identifiers. Although RAAQM can address the issue of uncontrollable RTT (Round-Trip Time) on different paths, its practical performance is not ideal due to the overhead of maintaining multiple cost information by the consumer. Apart from consumer-based statistical methods, there are also approaches that utilize routing nodes in the NDN network.

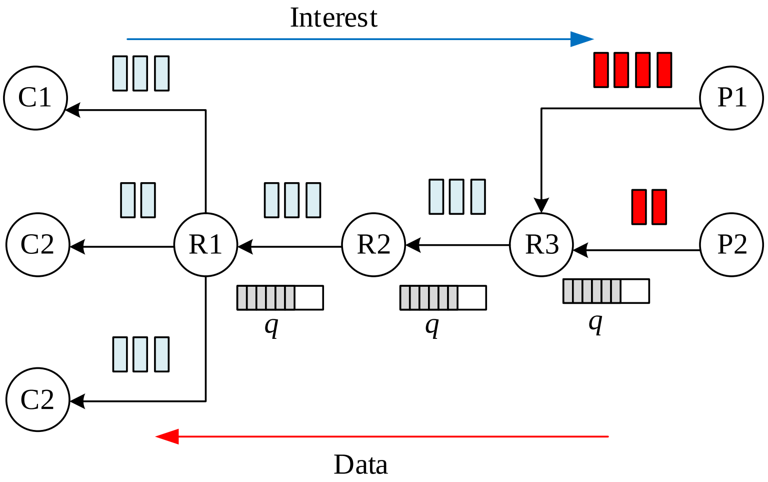


Figure 3. Transmitting procedure

For example, the ECP protocol [9] categorizes congestion levels into three levels and feeds back specific congestion level information to the receiver through explicit labeling. The receiver adjusts the congestion window (cwnd) using the MIAIMD algorithm. Another approach is the congestion control method based on VCP [10], which also considers link load. Intermediate nodes mark the congestion state based on the load factor. The consumer adjusts the interest packet transmission rate based on the received load factor. Figure 3 illustrates the process of active queue detection by intermediate nodes. Each cycle period T is evenly divided into N segments, where the duration of each segment (T/N) represents the probing period for the data queue length. Throughout the entire cycle period T, N+1 queue lengths q are calculated to update the load factor based on the changing trends of the queue length.

This type of receiver-driven congestion control protocol achieves congestion control by allowing the consumer to manage the transmission of packets, thereby reducing the influx of new traffic into the network from the source. However, most existing receiver-driven algorithms have a single congestion detection method and limited consumer control capabilities. They may not be able to respond promptly in the event of severe network congestion, leading to a significant number of packet losses.

3.2. Hop-by-Hop Congestion Control

The hop-by-hop congestion control algorithm is controlled by the intermediate nodes. When an intermediate node receives data, it adjusts its own forwarding rate based on its congestion condition, such as queue length, queuing delay, etc., using a congestion control algorithm [11].

In the Window-based protocol [12], the routing nodes multiply their own interest packet forwarding rate with the RTT of neighboring routers. This product is used as the optimal rate for each flow, effectively utilizing the network's bandwidth resources. The HIS protocol [13] considers that both interest packets and data packets can cause congestion and takes into account the correlation between them. The author utilizes mathematical analysis to determine the interest packet transmission rate, but the HIS protocol does not address flow fairness. The SF algorithm [14] sets rate limits for each interface, enabling more fine-grained restrictions on interfaces and content. The HoBHIS protocol [15] combines the queue utilization of different routers and the currently available resources to calculate the interest packet forwarding rate. Its working mechanism is illustrated in Figure 4, and the rate control is governed by Equation 2.

$$\gamma(t) = C(t) + h\frac{r - e(t)}{A(t)} \tag{2}$$

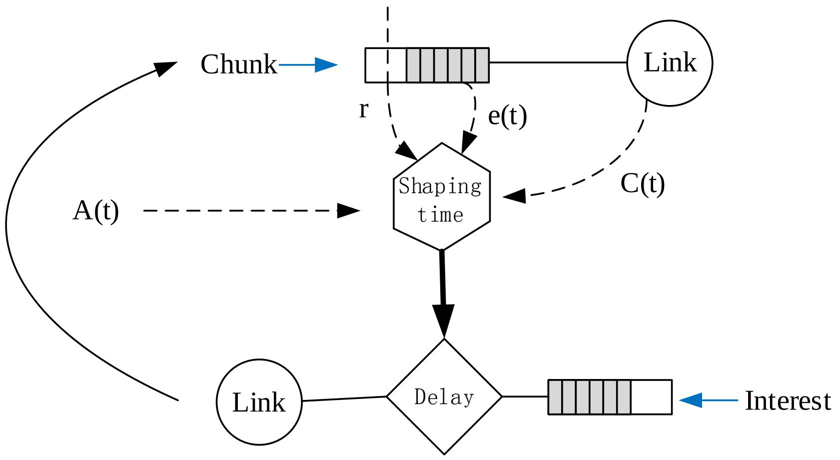


Figure 4. Working mechanism of HoBHIS

In the given context, $\gamma(t)$ represents the interest packet sending rate at time t, A(t) represents the response delay, h is a variable parameter, r represents the node's buffer capacity ratio, e(t) represents the utilization of the transmission queue, and C(t) represents the available bandwidth resources at time t.

Indeed, while hop-by-hop congestion control can respond more quickly to network congestion, a pure hop-by-hop mechanism is insufficient to guarantee optimal control performance. This is because when the receiver's sending rate cannot be adjusted in real-time based on the link status, hop-by-hop mechanisms cannot prevent new traffic from entering the network at the source to address congestion issues.

3.3. Hybrid Congestion Control

Hybrid Congestion Control is a new mechanism that combines the two aforementioned congestion control approaches. Compared to a single approach, the hybrid mechanism offers improved flexibility by incorporating control from both intermediate nodes and receivers, making it more suitable for congestion control in complex network environments [16].

The BCON algorithm [17] utilizes Active Queue Management (AQM) mechanism to manage congestion through queue-based management. When the intermediate node's pending packets exceed the queue length, a probabilistic packet dropping strategy is employed, triggering the NACK notification mechanism for each dropped packet. Upon receiving the NACK message, the receiver can reduce its interest packet sending rate to avoid further packet losses. The working mechanism is illustrated in Figure 5.

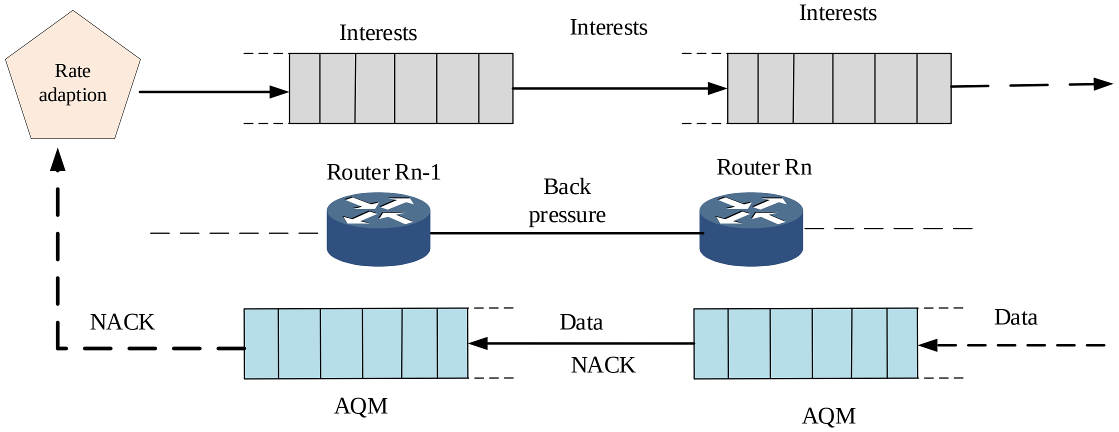


Figure 5. BCON control mechanism

The HR-ICP algorithm [18] incorporates hop-by-hop control into the ICP framework. The receiver sets a timer for each interest packet it sends, and if no corresponding data packet is received within the timeout period, it reduces the cwnd. The intermediate nodes allocate a virtual queue for each prefix and maintain a credit counter for each virtual queue. The credit counter helps estimate the downstream link rate information, which is used to adjust the forwarding rate.

The CHoPCoP algorithm [19] sets a queue length threshold at the routing nodes. When the queue exceeds the threshold, it triggers the fair sharing of interest packets mechanism for hop-by-hop control. Additionally, in cases of severe network congestion, the receiver also adjusts the congestion window. Other algorithms such as B-ICP [20] and Rate-based [21] operate by obtaining network status from intermediate nodes and adjusting the interest packet sending rate using either the receiver or the intermediate nodes.

These algorithms utilize the information from intermediate nodes to adjust the interest packet sending rate, either at the receiver or intermediate nodes, by acquiring network state information.

3.4. Other NDN Congestion Control Methods

The aforementioned three control approaches primarily rely on fixed rules as their design concept. The main limitation of this rule-based design is the inability to adjust congestion control strategies in a timely manner based on real-time network conditions. As a result, an increasing number of researchers are exploring the integration of machine learning, reinforcement learning, and network congestion control to make networks more intelligent.

In the traditional networking domain, Dong et al. proposed the PCC algorithm [22]. The PCC algorithm enables each sender to continuously observe the relationship between its actions and historical performance, allowing the sender to choose actions that sustain high network performance. Building upon PCC, Dong et al. further introduced the PCC Vivace algorithm [23], which incorporates online learning optimization algorithms to provide better assistance for network performance (throughput, latency, loss). Other algorithms include the QTCP algorithm proposed by Li et al. [24].

Although these algorithms cannot be directly applied to NDN, they provide new insights for many NDN researchers. For example, Narayan et al. proposed the CCP algorithm [25], which utilizes network state information (RTT, reception rate, loss rate, etc.) to control data transmission rates on multiple data paths to achieve congestion control. Additionally, Liu et al. developed the ACCP algorithm [26], and Jiayu Yang et al. proposed the IEACC algorithm [27]. Furthermore, Dehao Lan et al. introduced the DRL-CCP algorithm [28], which applies reinforcement learning and Q-learning to NDN networks.

4. SUMMARY

Currently, research on congestion control in NDN primarily focuses on receiver-driven congestion control algorithms, hop-by-hop interest shaping congestion control mechanisms, and network modeling and analysis related to congestion control. However, the inclusion of machine learning and reinforcement learning has brought new momentum to the development of NDN networks. In future research, a systematic and practical congestion control mechanism can not only perceive and regulate through intermediate routers and receivers but also continuously learn and improve from past experiences through reinforcement learning, machine learning, and other techniques, gradually moving towards optimal solutions. This would transform the entire NDN network from simply relying on fixed rules to a more intelligent network environment. Therefore, if such intelligent congestion control methods are implemented, research on congestion control mechanisms becomes more targeted when facing different NDN application scenarios, such as mobility, multi-homing, and multicast.

ACKNOWLEDGEMENTS

This work was supported in part by the Key Technologies R & D Program of Henan Province underGrant No.222102210049 and 242102211024, in part by the Longmen Laboratory Fronter.

REFERENCES

[1] Tariq A, Rehman R A, Kim B S. Forwarding strategies in NDN-based wireless networks: A survey [J]. IEEE Communications Surveys & Tutorials, 2019, 22(1): 68-95.

[2] Balador A, Cinque E, Pratesi M, et al. Survey on decentralized congestion control methods for vehicular communication [J]. Vehicular Communications, 2022, 33: 100394.

[3] Iqbal S , Hoque M, et al. A source-driven probabilistic forwarding and caching strategy in NDN and SDN-based NDN [J]. International Journal of Communication Systems, 2022, 35(6): e5093.

[4] Chaudhary P, Hubballi N, et al. eNCache: Improving content delivery with cooperative caching in Named Data Networking [J]. Computer Networks, 2023, 237: 110104.

[5] Donta P K, Srirama S N, Amgoth T, et al. iCoCoA: Intelligent congestion control algorithm for CoAP using deep reinforcement learning [J]. Journal of Ambient Intelligence and Humanized Computing, 2023, 14(3): 2951-2966.

[6] Mazziane Y B, Alouf S, Neglia G, et al. TTL model for an LRU-based similarity caching policy [J]. Computer Networks, 2024, 241: 110206.

[7] Carofiglio G, Gallo M, et al. ICP: Design and evaluation of an interest control protocol for content-centric networking [C]. 2012 Proceedings IEEE INFOCOM Workshops. IEEE, 2012: 304-309.

[8] Carofiglio G, Gallo M, Muscariello L, et al. Multipath congestion control in content-centric networks [C]//2013 IEEE conference on computer communications workshops (INFOCOM WKSHPS). IEEE, 2013: 363-368.

[9] Y. Ren, J. Li, S. Shi, L. Li, and G. Wang, “An explicit congestion control algorithm for named data networking,” in 2016 IEEE conference on computer communications workshops (INFOCOM WKSHPS). IEEE, 2016, pp. 294–299.

[10] Xing S, Yin B, Yao J, et al. A VCP-based congestion control algorithm in named data networking [C]. 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). IEEE, 2018: 463-468.

[11] Ye Y, Lee B, Qiao Y. Hop-by-hop congestion measurement and practical active queue management in NDN [C]//GLOBECOM 2020-2020 IEEE Global Communications Conference. IEEE, 2020: 1-6.

[12] Kato T, Bandai M, Yamamoto M. A congestion control method for named data networking with hop-by-hop window-based approach [J]. IEICE Transactions on Communications, 2019, 102(1): 97-110.

[13] Wang Y, Rozhnova N, Narayanan A, et al. An improved hop-by-hop interest shaper for congestion control in named data networking [J]. ACM SIGCOMM Computer Communication Review, 2013, 43(4): 55-60.

[14] Yi C, Afanasyev A, Moiseenko I, et al. A case for stateful forwarding plane [J]. Computer Communications, 2013, 36(7): 779-791.

[15] Rozhnova N, Fdida S. An effective hop-by-hop interest shaping mechanism for ccn communications [C]. IEEE INFOCOM Workshops. IEEE, 2012: 322-327.

[16] Li Z, Shen X, Xun H, et al. CoopCon: Cooperative hybrid congestion control scheme for named data networking [J]. IEEE Transactions on Network and Service Management, 2023.

[17] Agarwal A, Tahiliani M P. BCON: Back pressure based congestion avoidance model for Named Data Networks [C]//2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS). IEEE, 2016: 1-5.

[18] Carofiglio G, Gallo M, et al. Joint hop-by-hop and receiver-driven interest control protocol for content-centric networks [J]. ACM SIGCOMM Computer Communication Review, 2012, 42(4): 491-496.

[19] Zhang F, Zhang Y, Reznik A, et al. A transport protocol for content-centric networking with explicit congestion control [C]. International conference on computer communication and networks (ICCCN). IEEE, 2014: 1-8.

[20] Ye Y, Lee B, Flynn R, et al. B-icp: Backpressure interest control protocol for multipath communication in ndn [C]. IEEE Global Communications Conference. IEEE, 2017: 1-6.

[21] Zhong S, Liu Y, Li J, et al. A rate-based multipath-aware congestion control mechanism in named data networking [C]//2017 IEEE international symposium on parallel and distributed processing with applications and 2017 IEEE international conference on ubiquitous computing and communications (ISPA/IUCC). IEEE, 2017: 174-181.

[22] Dong M, Li Q, Zarchy D, et al. PCC: Re-architecting congestion control for consistent high performance [C]//12th Symposium on Networked Systems Design and Implementation. 2015: 395-408.

[23] Dong M, Meng T, Zarchy D, et al. PCC vivace: Online-learning congestion control [C]//15th Symposium on Networked Systems Design and Implementation. 2018: 343-356.

[24] Li W, Zhou F, Chowdhury K R, et al. QTCP: Adaptive congestion control with reinforcement learning [J]. IEEE Transactions on Network Science and Engineering, 2018, 6(3): 445-458.

[25] Narayan A, Cangialosi F, Raghavan D, et al. Restructuring endpoint congestion control [C]//Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication. 2018: 30-43.

[26] Liu T, Zhang M, Zhu J, et al. ACCP: adaptive congestion control protocol in named data networking based on deep learning [J]. Neural Computing and Applications, 2019, 31: 4675-4683.

[27] Yang J, Chen Y, Xue K, et al. IEACC: an intelligent edge-aided congestion control scheme for named data networking with deep reinforcement learning [J]. IEEE Transactions on Network and Service Management, 2022, 19(4): 4932-4947.

[28] Lan D, Tan X, Lv J, et al. A deep reinforcement learning based congestion control mechanism for NDN [C]//ICC 2019-2019 IEEE International Conference on Communications (ICC). IEEE, 2019: 1-7.