Computer Communications 86 (2016) 1–11

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Review

Congestion control in named data networking – A survey

Yongmao Ren[a,b], Jun Li[a,*], Shanshan Shi[a,c], Lingling Li[a], Guodong Wang[d], Beichuan Zhang[b]

[a] Computer Network Information Center, Chinese Academy of Sciences, P.O. Box 349,Beijing 100190, China
[b] University of Arizona, Tucson, AZ 85721, USA
[c] University of Chinese Academy of Sciences, Beijing, 100049, China
[d] South Dakota School of Mines and Technology, Rapid City, SD 57701, USA

ARTICLE INFO

Article history:
Received 9 September 2015
Revised 22 April 2016
Accepted 23 April 2016
Available online 25 April 2016

Keywords:
Named data networking
Congestion control
Information centric networking
Future Internet architecture
Transport protocol

ABSTRACT

As a typical Information Centric Networking, Named Data Networking (NDN) has attracted wide research attentions in recent years. NDN evolves today's host-centric network architecture TCP/IP to a data-centric network architecture. It turns the end-to-end connection-oriented transport of TCP/IP into receiver-driven connectionless transport. Compared with the traditional TCP/IP networking, the transport in NDN has new characteristics: *Receiver-driven, One-Interest-one-Data, Multi-Source, and Multi-Path.* These distinguished features pose new challenges to NDN congestion control mechanisms. This paper presents a comprehensive survey of state-of-art techniques aiming to address these issues, with particular focus on improving the effectiveness of congestion detection and the efficiency of Interest rate shaping. As a new research area, this paper also points out research challenges and open issues in this subject.

1. Introduction

Information-Centric Networking (ICN) has emerged as a promising candidate for the future Internet architecture. Named Data Networking (NDN [1,2]) is recognized as one of the most typical ICN architectures [3,4].

NDN was first proposed by Van Jacobson in 2009 [1]. Soon, it got positive responses from many researchers, as well as the NSF project support [5], and it has become a hotspot in the research of future Internet architecture. NDN rethinks and redesigns the current Internet architecture. The traditional Internet architecture is originally designed mainly for communication between exactly two machines. Most of the traffic on the Internet consists of connection-oriented TCP conversions between pairs of hosts. However, after a rapid development for more than half a century, great changes have taken place in the scale and applications of the Internet. Humans have entered into the era of "Big Data" and the main application mode has also been changed from text communication to information accessing and distribution. However, it is difficult for the current Internet architecture which is designed for the end to end communications to meet this change. Many problems emerge in terms of transport efficiency, security, mobility, etc [75–77]. Therefore, the NDN architecture is proposed to tackle these challenges by changing the focus of the Internet architecture from the current "where" (location) to "what" (content), centering on the named content (Data) rather than IP.

Compared to the traditional TCP/IP networking, NDN basically has the following new features [1,2,5]:

- It defines the name identifying content, and the routing is based on content name rather than IP address;
- It utilizes pervasive in-network caching to serve the requests by any potential content sources in the network rather than a single content source;
- It separates the routing and forwarding plane so that the forwarding strategy can be adjusted according to the data plane delivery feedback, allowing natural implementation of multi-path adaptive forwarding [6];
- It adopts the receiver-driven transport mode so that the data transfer can be controlled by adjusting request strategies at the receiver side;
- It secures the content itself rather than the communication channel [7].

Thanks to the above characteristics, NDN has many advantages in transfer efficiency [1], security [89] and mobility support [90] and some other aspects [91] compared to the traditional network. Therefore, it can better satisfy the requirements of the present and future applications.

Since the research on NDN is still in its initial phase, a number of specific technologies are still being studied, such as content naming [78], routing [79,80], transporting, content caching

* Corresponding author.
E-mail addresses: renyongmao@cstnet.cn (Y. Ren), lijun@cnic.cn (J. Li), shishanshan@cstnet.cn (S. Shi), lilingling@cstnet.cn (L. Li), Guodong.wang@sdsmt.edu (G. Wang), bzhang@cs.arizona.edu (B. Zhang).

http://dx.doi.org/10.1016/j.comcom.2016.04.017
0140-3664/

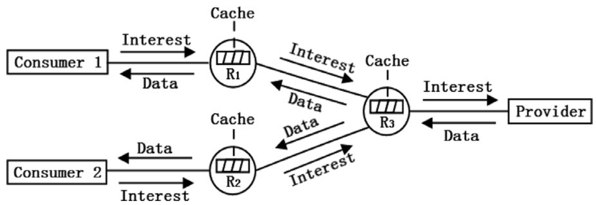

Fig. 1. NDN transport mode.

[48] and forwarding [29,81], security and privacy protection [82] . In this paper, we will analyze and summarize one of the key technologies, transport control technology especially the congestion control mechanism.

The contributions of this paper mainly include the following aspects.

- We identify the congestion control problems in NDN by analyzing the new features of the transport control in NDN, the problems of the traditional TCP[1] congestion control algorithms in NDN and the advantages of congestion control in NDN.
- We conclude and propose the methods of designing congestion control mechanisms for NDN, mainly including the receiver-based congestion control method, the hop-by-hop Interest shaping method, and the hybrid method. We also analyze the specific ideas of these methods and give typical examples. Moreover, we highlight the comparison among current typical schemes.
- We discuss and point out the research challenges and open issues for congestion control in NDN.

The rest of this paper is organized as follow. In Section 2, we introduce the transfer mode of NDN, compare the transport modes between NDN and the traditional TCP/IP networking, and point out the problems of using the traditional TCP congestion control algorithms in NDN and the advantages of the transport control of NDN itself. In Section 3, we analyze the methods of designing congestion control mechanisms for NDN and classify them into three categories: receiver-based control, hop-by-hop control and hybrid method. During the analysis, we discuss and compare the typical existing NDN congestion control algorithms and transport protocols. In Section 4, we point out the research challenges and open issues for congestion control in NDN, including the NDN transport model, congestion control mechanisms and performance metrics. Lastly, in Section 5, we conclude the whole paper.

2. Congestion control issues of NDN

2.1. NDN transport mode and new features

The transport control mechanism is one of the key technologies that need to be studied in NDN. The transport mode in NDN is quite different from the traditional TCP/IP network. As illustrated in Fig. 1, there are two NDN packet types, Interest and Data. Each NDN node (as shown in Fig. 2) has three main data structures: the Forwarding Information Base (FIB), which serves like IP FIB for routing, the Content Store (CS), which is used to cache content passing through, and the Pending Interest Table (PIT), which keeps track of Interests forwarded upstream toward content source(s) so that returned Data can be sent downstream to its requester(s)).

The basic communication process can be described as follows: Consumer 1 asks for content by sending out an Interest packet,

[1] There are many variants of TCP [12]. For simplicity and convenience, in this paper, when we mention TCP, it means the basic TCP version Reno TCP [97].

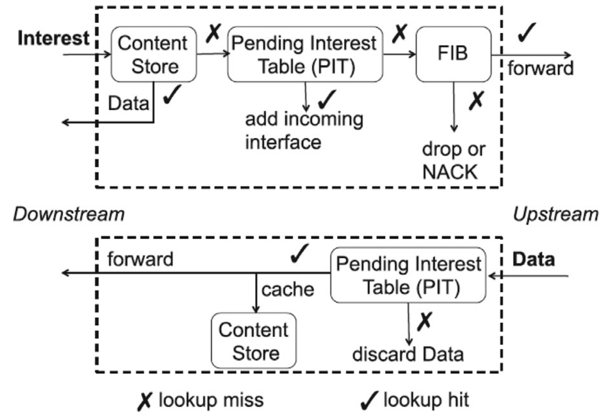

Fig. 2. Interest and Data processing in NDN node [6].

which carries a name that identifies the requested data. When received the Interest, the router R_1 first checks its CS, if there is already a Data packet in the CS that matches the Interest, R_1 will respond directly and send back the requested Data packet. Or else, it will record this Interest in the PIT table and forward this Interest by the FIB table. When the provider receives the Interest, it replies by sending back the corresponding Data. Data is not routed but simply follows the chain of PIT entries back to the original requester(s). While transferring Data, to maximize the probability of sharing, which minimizes upstream bandwidth demand and downstream latency, it will be cached in all the routers along the path as long as possible (Replacement algorithm is used, e.g., LRU or LFU). If Consumer 2 requests for the same Data, it can get it from the cache of router R_3 instead of the provider.

Compared to the traditional TCP/IP networking, transport in NDN mainly has the following new features:

- *Receiver-driven "Pull" mode*. A receiver requests content by sending Interest packets, and then, the data source nodes that satisfy the requirements return the requested data.
- *One-Interest-one-Data transport mode*. One Interest retrieves at most one Data packet. This basic rule ensures that flow balance is maintained in the network. Multiple Interests may be issued at once before Data arrives to consume the first Interest.
- *Multi-source transport*. As the transport of NDN is connectionless and content can be stored in network, a receiver can obtain the content responded by multiple content sources including the original content repositories and caches of intermediated nodes.
- *Multi-path transport*. NDN architecture intrinsically supports dynamic multi-path forwarding. The IP routing usually adopts a single best path to prevent loop[2]. The Multi-path TCP [33] only performs at the receiver over static paths. However, in NDN, routing is separated from forwarding, so it can adjust the forwarding strategy according to the delivery performance of data plane to make multi-path forwarding [29]. Interest loop is likely to be prevented, since the name plus a random nonce can effectively identify duplicates to discard [5]. On the other hand, Data do not loop since they take the reverse path of Interests. Thus multi-path communications can be safe of looping problems in NDN.

[2] There are some multi-path IP routing strategies like ECMP [92]. However, there may be significant problems in deploying multi-path routing in practice [92] and multi-path routing is not yet widely deployed in practice [93].

2.2. Problems of the traditional congestion control mechanisms in NDN

Since the NDN network transport is different from the traditional TCP/IP network transport, the traditional TCP congestion control algorithms and transport protocols cannot be directly applied in NDN network, mainly for the following problems:

- *Congestion Detection based on RTO is not reliable in NDN*
  Due to the new features of transport in NDN such as connectionless and multi-sources, the RTO (Retransmission Time Out) detecting mechanism based on a single-source connection (single path) in TCP is no longer reliable in NDN because the RTTs from different sources or different paths are different. Moreover, out-of-order delivery or changes of arriving time intervals may not be used directly as the indication of network congestion anymore, because the packet may come from different source nodes [8].
- *Congestion Detection Based on Duplicate ACKs is not suitable*
  Since NDN packets are independently named and one Data corresponds to one Interest, for the lost Data packets, it is the receiver's responsibility to retransmit the corresponding Interest requests, and the sender will not repeatedly send the same Data before receiving the same Interest request [5]. Namely, it does not need ordered packet transport and the transport does not stall on a loss. Therefore, the receiver cannot detect loss by receiving duplicate ACKs (Data packets[3]) like TCP. Therefore, the implicit congestion detection mechanism based on duplicate ACKs in traditional TCP is no longer suitable in NDN.
- *The rate control mechanism based on a single congestion control window is also not suitable in NDN*.
  TCP's congestion control window (cwnd) adjustment is aimed at a single path. It adapts to the bottleneck bandwidth of a single path by continually adjusting its cwnd size. However, in NDN, because of its multi-source characteristic [8,21,22], multi-path transfer is common, and the single cwnd control cannot adapt to multi-path transfer. For example, assume one consumer receives data from two sources through two different paths, if one path is congested and the consumer reduces its cwnd, it will also reduce the traffic of the other path which is not congested.
- *The self-clocking mechanism might cause the fairness problem between popular and unpopular contents*
  With TCP's self-clocking, the data-sending rate is determined by the rate of incoming ACKs. Competing TCP senders with different RTTs will receive ACKs at different rates and likewise will increase their sending window at different rates. This phenomenon is known as RTT-bias [98] of TCP. It was analytically derived in [32]. As a rule of thumb competing TCP flows attain throughputs inversely proportional to their RTTs. In NDN, with the existence of in-network cache, the popular content is cached closer to consumers and will be delivered with less delay. Thus, most link capacity will be occupied by the relatively more popular content, and only a small part of the available capacity will be claimed by the relatively less popular content, which leads to unfairness to the unpopular content.

In fact, the congestion control algorithm of TCP is also a continual research topic as the development and change of computer networks [12–17,26]. The congestion control in NDN network brings new challenges. Therefore, the existing congestion control schemes in TCP can only be studied for reference and cannot be used directly in NDN.

[3] From the viewpoint of congestion control, in NDN, the rate control is usually implemented by controlling the rate of Interest requests and Data packets can solve the role of ACKs like in TCP.

Additionally, although the UDP transport protocol of the traditional IP network is connectionless, it is unreliable and has no congestion control[4]. It is only suitable for some delay sensitive applications, and cannot be used on a large scale. Therefore, there is no congestion control mechanism of UDP which can be used or referred for NDN network.

2.3. Advantages of congestion control in NDN

On the other hand, the congestion control in NDN network also has its own features and advantages:

- *Capability of flow control at the receiver*. Because it adopts request-driven "Pull" transport mode and One-Interest-one-Data transport mode, Data traffic can be controlled by controlling the Interest traffic rate at the receiver. Congestion is mainly contributed by Data traffic since the size of Interest is relatively much smaller than Data [1]. So in this way, it may proact (and in cases prevent) congestion, while in a TCP/IP network, it directly controls the Data traffic which results in congestion. As a result, it can only make response after the happening of congestion. Therefore, it is an advantage of NDN compared with TCP/IP network.
- *Intermediate nodes can be used for congestion control*. In NDN, each PIT entry records an Interest packet that has been forwarded, waiting for the Data packet to return. The intermediate routers in NDN can manage the traffic load by managing the size of the PIT entry (the number of Pending Interests) [2,5]. It can control the rate of returning Data by controlling the rate of forwarding Interest. Therefore, it is easy for NDN to implement hop-by-hop congestion control [29].

Since the NDN network architecture is rather different from the traditional Internet architecture and the transport mechanism in NDN has its own features compared to the traditional networks, the congestion control algorithm in the traditional TCP/IP network is not applicable to the NDN network. Therefore, it is necessary to study the congestion control algorithm which is suitable to the NDN network.

3. Mechanisms for NDN congestion control

Since NDN uses the One-interest-one-data transport mode, Data packet traffic can be controlled by controlling the sending rate of Interest packets, thus to realize the flow control. On the other hand, by adjusting the sending rate of Interest packets, the loss of Data packet resulting from congestion may be prevented. Moreover, because an Interest packet is rather smaller than a Data packet, discarding an Interest packet in an early time will cost less waste of resources than lately discarding a Data packet. Thus, in NDN, we can control the rate of returning Data packets by throttling the request and forwarding rate of Interest packets as a means to realize in-network congestion control.

To design the algorithms of Interest rate shaping, based on the features of NDN transport, the congestion control mechanisms can be designed from the aspects of receiver or intermediate nodes. Therefore, according to the mode of congestion control, we divide these mechanisms into three categories:

- Receiver-based control
- Hop-by-hop control
- Hybrid method

Below we will analyze the design methods of congestion control mechanisms as well as the existing typical algorithms and protocols.

[4] Note that there are some proposed protocols which have congestion control on top of UDP, such as UDT [94], Rtsunami [95], and QUIC [96].

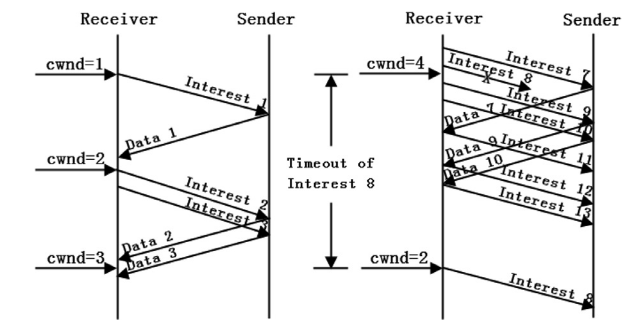

Fig. 3. Congestion window control at receiver: Increase (Left), Decrease (Right).

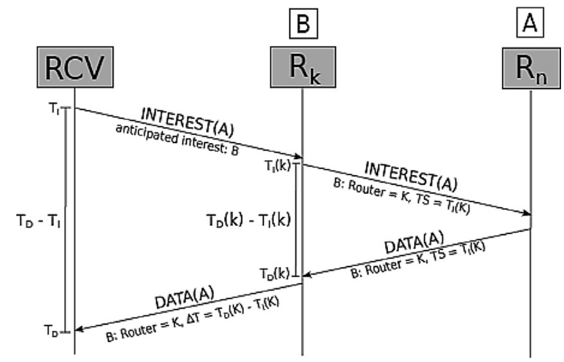

Fig. 4. Anticipated Interests mechanism [8].

3.1. Receiver-based control

Because NDN adopts the receiver-driven “Pull” mode, namely that the receiver requests content by sending Interest packets, and then, the data source nodes that satisfy the requirements return the requested data, one easy way to control the Data traffic is controlling the sending rate of Interest packets at receiver. Like TCP, window-based Interest transmissions can be used. But unlike TCP, here, only one type of window is needed, which serves the roles of both congestion window (the number of Interests allowed to be sent) and advertisement window (the number of Datas allowed to be sent), since NDN can control the Data traffic at the receiver. For the algorithm of adjusting window size, AIMD-like algorithm may be used.

Fig. 3 gives a simple illustration of this principle. The congestion control window (*cwnd*) is increased when a complete window of Interest is acknowledged by data reception (see the left figure). *cwnd* is decreased when a RTO timeout happens. At the same time, the unsatisfied Interest will be sent. (Note that it does not need to retransmit all the Interests in the window like TCP).

The detailed *cwnd* adjusting algorithm needs to be studied. Note that there is a different point: Since NDN packets are independently named and one Data corresponds to one Interest, it does not need ordered packet transport. NDN uses independent positive acknowledgement per request instead of cumulative acknowledgement of TCP. When there is an Interest packet lost, the sender will continue to send subsequent data packets corresponding to the Interests it received and the receiver will not receive duplicated Datas like TCP's duplicated ACKs. On the other hand, underlying packet networks might duplicate packets and NDN multipoint distribution may also cause duplication. Therefore, the receiver cannot detect congestion by duplicated Datas. The TCP's *Quick Recovery* algorithm based on duplicate ACKs is not applicable in NDN.

Another big difference is that, in NDN network, one content can have multiple data sources, since the content is usually multihoming and may be stored in the in-network caches. Therefore, the receiver also cannot detect congestion by single RTO timer. (It has been verified by simulations in [18].) The initial studies lack considerations on this issue. According to whether multi-source is considered, the existing control mechanisms based on receiver can be divided into Single-Source Algorithms and Multi-Source Algorithms.

3.1.1. Single source algorithms

One of the earliest Interest control protocols proposed for NDN is ICP [19] (Interest Control Protocol). ICP adopts the AIMD window adjustment algorithm like TCP as Eq. (1):

$$\begin{aligned} &\text{On Data}: w = w + \eta/W \\ &\text{On Loss}: w = \beta * w \end{aligned} \tag{1}$$

Where, the default value of η is 1, and $0 < \beta < 1$.

It detects congestion according to whether the RTO timer is timeout or not. The algorithm of setting the value of RTO timer τ is also similar with TCP as Eq. (2):

$$\tau = RTT_{\min} + (RTT_{\max} - RTT_{\min})\delta \tag{2}$$

Where $\delta = 0.5$, the initial value of $\tau = 10ms$ and $RTT_{\max} = 2RTT_{\min}$ initially after the first measured round-trip delay.

It sets RTO timer for each Interest packet, and measures the RTT for each returned Data packet. The congestion control window will be multiplicatively decreased when RTO timeout is triggered.

Another similar receiver-driven transport protocol proposed for NDN is ICTP (Information Centric Transport Protocol) [20]. It also adopts TCP-like window adjusting algorithm. It discusses the problem of proper size of data chunk to realize balance between safety overhead and flow control efficiency as well.

In NDN, the rate of arriving Data packets can reflect the network congestion status. Therefore, SIRC [53] proposes to control the sending Interest rate by adjusting the inter-Interest gap according to the arriving inter-Data gap. It also considers the switching of provider but without considering the transfer with multiple sources at the same time.

3.1.2. Multi-source algorithms

One possible solution to solve the multi-source problem is maintaining a separate RTO value for each content source. A simple implementation of this solution is ConTug [21]. However, ConTug assumes that the receiver knows the location of each content chunk before the transfer starts and the location does not change during the flow transfer, which is not currently possible in NDN and is not easily achievable without compromising scalability [8]. More practical solutions can be classified to the following three main methods.

(1) One RTO timer for each source

A possible solution to solve the multi-source problem is maintaining a separate RTO value for each data source, which is used in CCTCP (Content Centric TCP) [8]. To solve the multi-source issue, CCTCP maintains multiple congestion windows and RTO values for each flow: one corresponds to each cache. Because the receiver maintains multiple RTO values for each flow, thus, a mechanism is needed to estimate where the packet comes from so as to trigger the correct RTO. Therefore, CCTCP proposes an “Anticipated Interests” mechanism to reliably predict the location of chunks, then exactly estimate the retransmission timeout before chunks are requested, and it maintains a separate RTO for each anticipated source.

The basic rationale is shown as Fig. 4. The receiver sends Interest carrying the information of the subsequent expected Interest. When the intermediate node *k* receives the Interest, if the cache of this node has the data corresponding to the subsequent Interest, the identifiers of this node and the chunks as well as the timestamp will be appended to this Interest. When the receiver receives the Data, it will calculate the RTT(k) between itself and the node k with expected content according to the information of timestamp by Eq. (3):

$$RTT(k) = (T_D - T_I) - (T_D(k) - T_I(k)) \tag{3}$$

Where, T_I and $T_D$are measured when the receiver sends Interest packages and receives Data packages respectively. $T_D(k) - T_I(k)$ is measured by node k. Based on this value, the receiver sets the expected RTO of the Interest.

CCTCP's rate adjustment algorithm is derived from TCP New Reno, and comprises slow start and congestion avoidance phases. But its rate control is implemented at receivers through the regulation of Interest packets. Furthermore, a receiver may keep multiple congestion windows for each flow.

(1) One RTO timer for each path

An alternative solution is maintaining a separate RTO value for each transfer path. A representative scheme is RAAQM (Remote Adaptive Active Queue Management) algorithm [22].

The rationale is that each Data packet is appended with the identifier of the intermediate node while passing by. Then a Route Label is composed by the identifier sequence of traversed nodes, and the receiver detects RTO timeout by each route to realize congestion control of multi-source and multi-path. The congestion detection uses the algorithm similar to FAST TCP [23] protocol, and it estimates the network congestion levels according to the changes of RTT measured, and then calculates the corresponding probability of reducing *cwnd*. This probability value has a linear relationship with the real-time value of RTT on an interval $[R_{min}\ R_{max}]$ as calculated by Eq. (4):

$$p(t) = p_{\min} + \Delta p_{\max} \frac{R(t) - R_{\min}(t)}{R_{\max}(t) - R_{\min}(t)} \tag{4}$$

Where, $\Delta p_{\max} = p_{\max} - p_{\min}$, *R(t)* is the real-time value of RTT of the currently sent Interest package, $RTT_{\min}(t)$and$RTT_{\max}(t)$ respectively means the minimum and maximum of the RTT estimated based on historical samples to time t. When$RTT_{\max} = RTT_{\min}$, $p = p_{\min}$.

(1) One RTO timer for each CS

Another possible solution is maintaining a separate RTO value for each CS, which is proposed in [18]. Content packets are marked and handled with a unique per-CS identifier. The receiver does accounting per CS identifier it has seen as before and stores for each CS the content chunk ranges it has cached. Based on this information, it maintains distinct RTO estimators per CS.

These above solutions consider the multi-source issue of the transport in NDN, and transform the multi-source problem into multiple single-source problems by identifying each content source or each transport path. However, they still have two problems. The first is the problem of reliability. These solutions still use the traditional TCP's implicit congestion control mechanism (estimating network congestion according to the RTO timeout or the changes of RTT), but the methods of indirect estimation can not distinguish the timeout caused by the loss of packet and congestion, and they can not accurately reflect the network congestion levels (It has long been criticized [24,25]). The second is the high cost because the receiver needs to maintain much information of multiple sources or paths for each flow (such as multiple RTO values).

3.1.3. Explicit control algorithms

Different from the above implicit congestion control methods, a new direction to solve the multi-source problem is using explicit congestion detection and notification method. The basic idea is that the receiver adjusts its Interest sending rate according to the explicit congestion information detected and fed back by the routers. Fig. 5 illustrates a typical explicit congestion control model in NDN. The intermediate router nodes (such as the nodes of the bottleneck link) periodically detect its congestion status (usually by detecting the average queue length). If congestion happens, the congested router will feed back the congestion information to down stream nodes. Generally, there are two kinds of feedback methods. One method is piggybacking the congestion information by Data packet by using special bit field. This method likes the ECN [27,28,74] notification method proposed in the traditional IP network. The other method is using special control signal packet like the NACK [29] proposed in NDN network. The marked Data packet or the special NACK packet is forwarded downstream along the Data forwarding path (namely the Interest request path recorded in PIT table) to the receiver. While passing through each intermediate node, the congestion status information will be updated through comparison between the original congestion level in NACK and the level detected in this node itself. Once receiving the explicit congestion information, the receiver will adjust its Interest sending rate correspondingly. By this way, the receiver does not need to "guess" (implicitly estimate) the network congestion status any more. Therefore, the issue of different RTTs from different sources is not a problem any more.

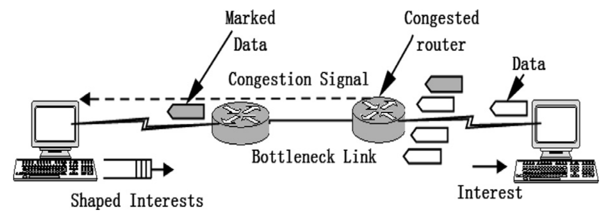

Fig. 5. An explicit congestion control model.

An attempt of this method is made in [30]. This scheme is called CCS (Congestion Control Scheme). It uses the RED (random early detection [31]) algorithm to detect the congestion levels and mark the congestion information in the returning Data packets. The receiver uses a kind of AIAD (Additive Increase Additive Decrease) algorithm to adjust the Interest sending rate according to the feedback congestion levels. Another explicit scheme is the ECP (Explicit Control Protocol) protocol [34,99]. It divides the congestion degree into three levels and explicitly feeds back the specific congestion level information to the receiver, and then the receiver adjusts its Interest sending rate by a MIAIMD (Multiplicative Increase Additive Increase Multiplicative Decrease) algorithm correspondingly. Another ECN-based scheme SECN [72] is proposed. It explicitly notifies the receiver of the available bandwidth and adopts a SDN-style (Software Defined Networking) controller to select the best forwarding path. It needs all the receivers send their RTTs to the routers for calculating the available bandwidth and it needs a centralized controller to select paths.

Additionally, the neural network algorithm is proposed as an explicit method to detect the congestion [35,36,84]. But this method is too complex and many parameters are hard to dynamically gather. The explicit congestion control method is a good way to solve the multi-source problem in CCN, and it needs further research.

3.2. Hop-by-hop Interest shaping

In TCP/IP network, the intermediate nodes are responsible for forwarding packets and the congestion control is mainly the business of terminals. However, in NDN network, the intermediate nodes can use PIT tables to record pending Interests and can

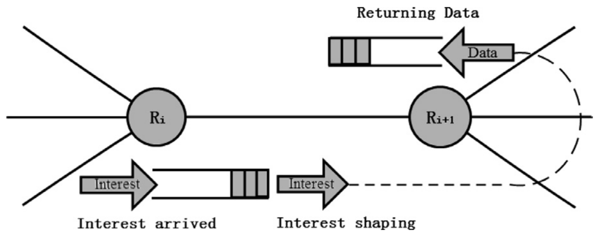

Fig. 6. Hop-by-hop NDN congestion control.

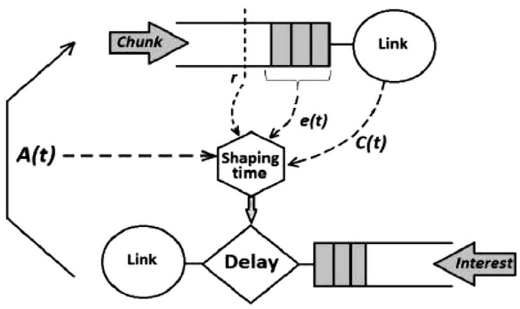

Fig. 7. HoBHIS node rate adjustment model [37].

shape Interest forwarding rate. Therefore, another congestion control mode which is different from Receiver-based control in NDN is hop-by-hop control. Some algorithms have been proposed in [37,29,38,73,83].

The basic principle of hop-by-hop control is as follows: Each NDN node detects congestion and adjusts the forwarding rate of Interest packets to control the transfer rate of returning Data correspondingly, thus to realize network congestion control. The basic idea is illustrated as Fig. 6.

Generally, the intermediate node Router i (R_i) detects its own congestion status by detecting the arriving data queue length, if it is congested, it will reduce the rate of forwarding Interests. The specific congestion detection and Interest shaping algorithms still need to be studied.

A typical hop-by-hop Interest rate adjustment algorithm is named HoBHIS (Hop-By-Hop Interest Shaping) proposed in [37] [73]. It detects congestion by monitoring the chunk queue length and adjusts the Interest forwarding rate according to the queue occupancy and available resources. The process of rate adjusting at nodes is illustrated as Fig. 7.

Upon arrival of a Data chunk in the transmission queue, the router computes the Interest rate $\gamma(t)$ based on the queue occupancy *e(t)* and the available resources (available bandwidth *C(t)* and free queue length during one Response Delay *A(t)* namely one RTT) for each conversation as Eq. (5). If the queue length is less than some threshold value r, then the router can temporarily increase the shaping rate. Otherwise, the router will decrease its shaping rate.

$$\gamma(t) = C(t) + h\frac{r - e(t)}{A(t)} \tag{5}$$

Where, *C(t)*: available bandwidth, r: threshold of data chunk queue, *e(t)*: the queue length of data chunk at time t, *A(t)* is the Response Delay, namely RTT, which can be measured at the node. *h* is a control parameter.

In addition to shaping the Interest forwarding rate of itself, the intermediate node $R_i$can also notify its downstream nodes R_{i-1} of its congestion status. An NACK (Negative ACK) feedback and multi-path forwarding mechanism is proposed in [29]. A congestion NACK is generated and fed back to downstream node if the Interest cannot be forwarded upstream due to congestion. The downstream node will try other interfaces for this Interest. It uses a “Three Colors” algorithm to mark and rank the interface performance, and the node can make adaptive multi-path forwarding according to the interface performance so as to mitigate congestion.

Usually, only the Data packets are considered for contribution to congestion because the size of Interest packets is small. For further research, the Interest packets can also be considered. For example, [38] considers the Interest packet size, and thinks that the Interest packets also contribute to the link congestion. Based on the constraint condition that the sum of Interest packet rate and Data packet rate should be less than the bandwidth, the optimal Interest shaping rate is computed. It detects the incoming Interest rate at intermediate nodes, and then adjusts the Interest forwarding rate based on the difference between the expected optimal rate and the actual incoming Interest rate. The shaping rate is calculated by Eq. (6):

$$\min\_s_1 + (\max\_s_1 - \min\_s_1)\left(1 - \frac{obs\_s_2}{\exp\min\_s_2}\right)^2 \tag{6}$$

Where, obs_s_2 is the detected actual incoming Interest package rate, $expmin\_s_2$ is the expected optimal incoming interest, and the Interest package rate s_1 is adjusted according to the difference between min_s_1 and max_s_2. (For convenience, we named this algorithm HIS hereinafter.)

The hop-by-hop congestion control mode adapts to features of connectionless and multi-source of the NDN network transport. Therefore, it is a very promising research direction. The research on this field is currently insufficient. Most of current existing hop-by-hop NDN congestion control algorithms rely on bandwidth value and simply suppose the bandwidth is known and constant. Thus, the routers use its interface bandwidth as the available link bandwidth between two NDN nodes to compute the forwarding Interest shaping rate. In fact, this is not practical. First, the available link bandwidth is the interface bandwidth only if the two nodes are directly connected by a dedicated physical link. However, there are many cases that the two NDN nodes are connected by a shared link, or an overlay link across a number of IP routers and/or Ethernet switches. Moreover, the available bandwidth is not constant, for example, the link bandwidth is dynamically changing for wireless link like Wifi. Therefore, this is a big problem of current hop-by-hop congestion control algorithms in NDN. Additionally, further studies in terms of the accuracy of congestion detection, the calculation of fair rate, etc., are still needed. Besides, a point which has not been considered is the cost. For example, the complex processing on each intermediate node will bring high overhead, the cost and its effects on the forwarding performance still need to be studied.

3.3. Hybrid method

The pure Receiver-based control mode has the problem of accuracy in congestion detection. Moreover, in terms of fairness, if the rate control mechanism is only implemented at terminal, when there is a greedy terminal which uses other rate control mechanism like constant Interest sending rate rather than the mechanism similar to TCP, the bandwidth will be occupied by it. In this case, control at intermediate nodes has advantage since it can regulate all flows[5] (for example, it can regulate flows according to the fair rate of each flow). On the other hand, the pure hop-by-hop mechanism without receiver-based rate control is not enough to guarantee the optimal transport performance, because the optimal initial

[5] NDN flows can be identified on-the-fly from the object name included in Interest and Data packets [66].

sending rate of the terminal is not a priori value and it changes according to the variation of available network resources [39]. Therefore, the combination of the two mechanisms may be a congestion control method.

An existing exploration of the hybrid method is HR-ICP (Hop-by-hop and Receiver-Driven Interest Control Protocol) [39]. It adds the hop-by-hop congestion mechanism based on the ICP. At receiver, it uses ICP's Interests control algorithm, and at intermediate nodes, it uses the following algorithm.

In each output interface, HR-ICP maintains one virtual queue for each flow and associates a credit counter (initialized to a maximum value B, indicating the number of Data bytes that the flow is granted to transmit). The counter is increased according to the estimated fair rate of corresponding down-link and decreased by the number of forwarded Interests. A flow refers to a single content retrieval, and is defined as bottlenecked when at least one Interest packet is queued at the output buffer or the credit counter is null. The life period of virtual queue is the same with that of related Interest packets in PIT. When an Interest packet arrives at an interface, the node checks if it belongs to a bottlenecked flow or not. If the flow is not a bottlenecked flow, the Interest will be directly forwarded out and the credit counter is decreased by the number of bytes of the corresponding Data packet. For a bottlenecked flow, the Interest packet will be queued in a drop tail FIFO of size Q_{max} for rate adapting. Suppose $\phi_i(t)$ is the shaping rate of node i at time t, and, as illustrated in Eq. (7), it is calculated by that the link capacity C_{i+1} minuses the total rate of non-bottlenecked flows, then the remaining value is equally divided by the number of bottlenecked flows *N*.

$$\phi_i(t) = \left(C_{i+1} - \sum_{j=1}^{M} X(j,t) \right) / N \tag{7}$$

Where, *X(j,t)* is the rate of non-bottlenecked flow *j* at time *t*, and *M* is the number of non-bottlenecked flows. The results of simulations verified that HR-ICP can fast detect and respond to the congestion state, thus to make fast convergence and improve throughput and robustness, compared to ICP, which only adopts receiver-based control mechanism.

In HR-ICP, the receiver-based mechanism and the hop-by-hop control mechanism are independent and there is no collaboration. To enhance the collaboration between the two mechanisms, strategies need to be studied. For example, we may set some conditions to start each control. Another hybrid control algorithm named CHoPCoP (Chunk-switched Hop Pull Control Protocol) proposed in [40,85] sets a threshold of queue length in intermediate node. It starts the hop-by-hop control by its FISP (Fair share Interest shaping) mechanism only if the queue length is longer than this threshold, which means that the receiver-based control is not enough to effectively prevent congestion. By this way, CHoPCoP uses the receiver-based control as the main control, and uses the hop-by-hop control to assist it to realize the whole congestion control. Besides, except for using AIMD window adjusting mechanism at receiver, CHoPCoP also proposes using REM (Random Early Marking) [41] mechanism in NDN. The intermediate nodes detect congestion by monitoring the size of outgoing data queue, and explicitly marks data packets to notify receivers when the network is congested. This explicit congestion signaling instead of RTT-based congestion prediction can directly let the receivers know the congestion status.

In hybrid control, for control at the receiver, the problems as we described at Section 3.1 like multi-source issue still need to be considered. Based on RAAQM [22], which maintains a separate RTO value for each transfer path, [42] adds control at intermediate nodes. It uses multi-path forwarding at intermediate nodes by giving each output interface weight for each incoming Interest to choose forwarding path.

When using the hybrid method, how to coordinate between the control on receivers and the control on intermediate nodes still needs to be studied.

3.4. Comparison of existing congestion control mechanisms for NDN

We classify the existing NDN network congestion control mechanisms based on the above analysis and discussion, and compare them from three aspects: congestion control model, congestion detection method and rate adaptation algorithm, as shown in Table 1.

Generally speaking, the receiver-based control mode implements congestion detection and rate adaptation only at the receiver, which matches the receiver-driven "Pull" transport mode of NDN and is easy to be realized. However, it is location related, and needs to solve the problem of accurately detecting congestion from different content sources. The existing multi-source algorithms can mitigate this problem but their costs are high. The explicit congestion control method can detect and notify the congestion status correctly and timely, but if it is a pure receiver-based method, a single congestion window may be hard to adapt the complex congestion status when transporting in multi-path.

The hop-by-hop Interest shaping mode detects congestion and controls rate at each intermediate node, which can detect congestion more accurately and adjust rate more carefully than the receiver-based control mode. It does not care the location of each receiver and content source, which matches the "Sessionless" spirit of NDN architecture design. However, handling congestion at every intermediate node is complex and high cost.

4. Research challenges and open issues

Since the NDN network architecture is a new future Internet architecture, it has many different features compared to the traditional network architecture. Moreover, the transport mode and mechanism of NDN and that of the traditional network are very different. Therefore, a vast amount of further researches are needed for the transport mechanism of NDN network. Although some of the existing studies have made useful explorations, there are still a lot of issues needing to be addressed. We list some important yet challenging problems here.

4.1. NDN network transport model

The network transport model is the base of designing a congestion control mechanism. Different transport models require different corresponding congestion control schemes. The transport model of the NDN network is very different from that of the traditional TCP/IP network, especially on that the in-network cache is introduced as a major innovation in NDN, which will make big changes on transport model.

Constructing NDN network transport model helps to reveal the features of NDN network transport, as well as find the key factors that affect the transport performance and their relationships, and lay the foundation for designing congestion control algorithm with high performance. There is already some work [43–47]. An analytical model of ICN storage and bandwidth sharing is developed in [45], which derives an expression for expected stationary delivery time as a function of hit/miss probabilities at network caches, content popularity and cache sizes. Compared to the studies on congestion control algorithms of NDN, the studies on transport model of NDN are relatively few. More work needs to be done, such as the establishment of more sophisticated caching network analysis model of NDN network [48], analysis on the relationship between transport performance indicators (such as throughput, delay and

Table 1
Comparison of existing congestion control mechanisms for NDN.

Control modes			Protocols	Congestion detection	Interest rate shaping
Receiver-based control	Implicit congestion control	Single source	ICP [19], ICTP [20]	RTO timeout	AIMD-like end-to-end rate adjustment
			SIRC [53]	RTO Timeout	Adjust inter-Interest gap according to arriving inter-Data gap
		Multi-Source	ConTug [21]	One RTO timer per source; Assume the location of source is known.	Each CWND for each source, AIMD-like rate adjustment but linearly decrease on weak congestion
			CCTCP [8]	One RTO timer per source; Measure each RTT of each source.	Each CWND for each source, AIMD-like rate adjustment.
			RAAQM [22]	One RTO timer per path; Measure each RTT of each path.	Each CWND for each path; The reducing probability changes with RTT.
			"predictive" [18]	One RTO timer per CS; Measure each RTT of each CS.	AIMD algorithm
	Explicit Congestion Control		CCS [30]	RED-like algorithm	AIAD-like end-to-end rate adjustment
			ECP[34]	Length of Interest queue	MIAIMD algorithm
			SECN [72]	Router computes available bandwidth	Controller chooses the best forwarding path
Hop-by-hop Interest shaping			HoBHIS [37]	Chunk queue length	According to available bandwidth and queue.
			HIS [38]	Incoming Interest Rate	According to bandwidth and the traffic of both Data and Interest.
			NACK [29]	Outing Interest Queue	Multi-path forwarding
Hybrid method			HR-ICP [39]	Intermediate nodes: compare Interest rate and its associated Data Rate Receiver: RTO	Intermediate nodes: according to fair rate of each flow; Receiver: AIMD.
			CHoPCoP [40,85]	Intermediate nodes: Detect outgoing Data queue length and notify receiver by REM	Intermediate nodes: If queue length > Threshold, delay Interest forwarding; Receiver: AIMD.

packet loss rate) and factors like caching strategy (includes cache location, cache size, cache replacement algorithm, etc.), content popularity, content size, etc. In addition, not only the in-network cache, but also other features of NDN like multi-source, multi-path transfer modes need to be studied in NDN network transport model. Current designs of congestion control algorithm lack deep analysis of the features of the NDN transport model, and more integration should be done in future.

4.2. Congestion control mechanisms for NDN network

As we pointed out at Section 2, the TCP transport protocol can not be simply used in NDN network and the congestion control algorithms suitable for transport in NDN need to be studied. The research in this field is still in the initial stage, challenges and further research directions can be mainly classified as follows:

(1) Hybrid congestion control mechanisms

The existing receiver-based control mechanisms and protocols have drawbacks such as: without consideration on the multi-source issue, or the accuracy of congestion detection and the cost issue in transforming the multi-source and multi-path problems to a single source and single path problem simply by maintaining one RTO timer for each source or path. The hop-by-hop congestion control mode adapts to the features of connectionless and multi-source of NDN transport. However, because congestion control is done on every node, its cost is high. The method of hybrid control mechanisms may make full use of the advantages of them, and it's needed to study how to combine and coordinate these two mechanisms and avoid the transport performance degradation caused by the repeated and excessive control.

(2) Congestion control mechanisms considering cache

Cache has important influences on the transport traffic. In network storage is one of the key features of NDN [48,49]. Therefore, there is very close relationship between caching strategy and congestion control mechanism. Currently, most of the existing congestion control mechanisms have not considered cache. How to design congestion control mechanism for NDN considering in network storage is a big open issue. On the other hand, when designing caching strategy, the flow and congestion control should also be considered. There are already some preliminary works [50,51]. Additionally, integrating dynamic forwarding and caching is also a way to achieve network congestion control and load balancing [52].

(3) Congestion control with multi-path transport

The capability of multi-path transport is also one of the key advantages of NDN [29]. When continual heavy congestion happens, using rate control on one path only may be not enough to mitigate congestion. In this case, multi-path transport may be a good way to distribute traffic and mitigate congestion. Therefore, design congestion control mechanism integrating with multi-path transport strategy is a promising direction to realize congestion control in NDN. Some preliminary works can be referred to [29,42,52,86,87].

(4) Congestion control in wireless scenarios

The current studies on the congestion control in NDN mainly focus on the wired network and the studies on the wireless network environment are very lack. In NDN's architecture, there is a special "Strategy" layer, which can make full use of all kinds of under layer links including different wireless links. The capability of multi-path transport also makes NDN different from the traditional IP network in wireless environment. Except for these advantages, there are also some new challenges for running NDN network in wireless environment. For example, the communication in NDN can dynamically vary from one-to-one to one-to-many nodes, due to content multi-homing at various caching points [18]. The impact of multi-homed content is further exacerbated in a wireless environment due to nodes mobility: closer content stores may start serving suddenly the consumer, while other content stores could become suddenly unreachable [53]. Therefore, because of the advantages and challenges of NDN transport in wireless environment, the congestion control mechanisms of NDN in wireless environment need to be studied. Some research exploration on this field can be referred to [53–57].

(5) Congestion control aimed at specific applications

Except the congestion control of NDN transport in different environments, the congestion control of NDN transport in different specific applications also needs to be studied. For example, the congestion control schemes for video streaming transfer over NDN [58,59]. Additionally, different applications have different transfer quality requirement. When congestion occurs, their traffic can be controlled by different rate control schemes [88].

(6) Retransmission and reliability

Besides, the current researches on the congestion control in NDN mainly focus on the congestion detection, notification and Interest rate shaping, and there are few studies on the follow-up work after congestion like retransmission. This is in fact another important topic, namely the transport reliability. Because both interest packets and data chunks can be dropped on the way, due to network congestion or impairment, deciding who, the NDN router or the end-system, sets the timeout duration and retransmits the interest packet after a timeout is the key issue that affects the performance of the network. More specifically, it impacts directly the occupancy of the pending interest table (PIT) [60]. Therefore, this is also an open issue. Some preliminary works can be referred to [60–65]. For example, a subscriber feedback mechanism named R-COPSS is proposed in [64] to reduce traffic from the aspect of retransmission and reliability. It can realize congestion control and fairness across publishers especially in multicast scenario.

Except the above five directions of congestion control mechanism for NDN network, there are also some other research issues and directions. For example, the Interest aggregation mechanism [69] and the redundancy elimination mechanism [70] can improve the transport efficiency and reduce congestion.

4.3. Fairness and TCP-Friendliness of NDN network congestion control algorithm

Fairness is an open issue for NDN. There are many problems needed to be solved. For example, firstly, how to define the concept of fairness in a NDN network? The traditional fairness is usually defined in terms of flow, although there are many kinds of fairness indexes defined. However, in NDN, the concept of flow is different from TCP/IP network, because in NDN, we do not care about the location of content source, so content may come from multiple sources by multiple paths. Therefore, the concept of flow should be redefined. Moreover, in NDN, the fairness may also be evaluated from different aspects, such as flow fairness, content fairness, link/interface fairness and user fairness. Each kind of fairness needs further research. For example, for content fairness, what granularity of fairness should be used is an issue, such as per-file, per-content and per-chunk. Also, how to measure fairness needs to be studied. In [66], per-flow fairness of NDN is discussed and a control framework and some mechanisms are proposed for reference.

For congestion control algorithm design, as discussed at Section. 2.2, the self-clocking mechanism might cause the fairness problem between popular and unpopular contents. In NDN network, the content with high popularity is cached closer to the consumer, and it has smaller RTT delay. If we simply use the same self-clocking mechanism as TCP, it will result in that most of the link capacity is occupied by the content with high popularity, and only little link capacity is available for the content with low popularity, which is unfair for low popularity content [9–11]. Besides, when designing the deployment of content with different popularity in cache, fairness should also be considered [67].

Therefore, research on the fairness of congestion control algorithm in NDN network is needed. However, it is absent in most existing NDN network protocols. Therefore, it is necessary to consider the fairness and optimize these congestion control algorithms. In the future Internet environment, NDN and TCP/IP architectures may coexist, and then the issue that whether NDN network transport protocol is compatible to TCP transport protocol needs to be studied. [68] makes some preliminary research on the TCP-friendliness of flow control mechanisms adopting timeout driven AIMD strategy in NDN.

The performance evaluation of congestion control algorithms for NDN network is still an open issue. Performance comparison among the algorithms proposed for NDN congestion control is necessary. Performance metrics include throughput, content delivery time, fairness, TCP-friendliness, etc. There are already some preliminary works, like the results reported in [71].

In conclusion, as a new architecture, NDN has many challenges and open issues for its congestion control. A key advantage of NDN compared to IP network is that users can get content from the caches of near routers, which reduces redundant transfer tremendously and improves the transfer performance obviously. The new multi-source and multi-path transport mode improves the transfer performance vastly. At the same time, it brings a new big challenge for the network congestion control. A good congestion control algorithm for NDN has to work well in the multi-source and multi-path transfer scenarios.

5. Conclusion

There are significant differences between the NDN architecture which is centered on content and the traditional TCP/IP architecture which is centered on IP address, and their transport modes are also very different. Therefore, the congestion control mechanisms which are studied and designed in traditional TCP/IP networks cannot be directly adopted in NDN network. The new congestion control mechanisms and transport protocols of NDN network need to be studied. In this paper, we firstly highlight several new features of NDN transport, and point out the congestion control issues in NDN. Then, we analyze the methods of designing congestion control mechanisms for NDN by classifying them into three categories: receiver-based control, hop-by-hop control and hybrid method. The typical existing NDN congestion control algorithms and transport protocols are also classified and analyzed with emphases on their principles and comparison. Based on the deep analysis and discussion in this paper, research challenges and open issues in this field are proposed, which is helpful for researchers to quickly involve in and move forward.

Acknowledgment

This work has been supported by the Around Five Top Priorities of “One-Three-Five” Strategic Planning, CNIC (Grant No. CNIC_PY-1401), the National 973 program (Grant No. 2012CB315803) and the China Scholarship Council. The authors would like to acknowledge the reviewers and editors for their through and expert assistance in improving the quality of the paper.

References

[1] V. Jacobson, D.K Smetters, J.D. Thornton, M.F. Plass, N.H. Briggs, R.L. Braynard, Networking named content, Proceeding of CoNEXT'09, 2009.
[2] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, K. claffy, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang, Named data networking, ACM SIGCOMM Comput. Commun. Rev. 44 (3) (2014) 66–73.

[3] G. Xylomenos, C.N. Ververidis, V.A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K.V. Kstsaros, G.C. Polyzos, A survey of information-centric networking research, IEEE Commun. Surv. Tut. 16 (2) (2013) 1024–1049.
[4] G. Xie, Y. Zhang, Z. Li, Y. Sun, Y. Xie, Z. Li, A survey on future internet architecture, Chinese J. Comput. 35 (6) (2012) 1109–1119.
[5] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J.D. Thornton, D.K. Smetters, Named data networking (NDN) project, NSF Project Proposal, 2010.
[6] C. Yi, A. Afanasyev, L. Wang, B. Zhang, L. Zhang, Adaptive forwarding in named data networking, ACM SIGCOMM Comput. Commun. Rev. 42 (3) (2012) 62–67.
[7] X. Zhang, K. Chang, H. Xiong, Y. Wen, G. Shi, G. Wang, Towards name-based trust and security for content-centric network, Proceeding of the 19th IEEE ICNP, 2011.
[8] L. Saino, C. Cocora, G. Pavlou, CCTCP: a scalable receiver-driven congestion control protocol for content centric networking, Proceeding of IEEE ICC'13, 2013.
[9] D. Saucez, L.A. Grieco, C. Barakat, AIMD and CCN: past and novel acronyms working together in the Future Internet, Proceeding of ACM CoNEXT 2012, CSWS Workshop, 2012.
[10] M. Becke, T. Dreibholz, H. Adhari, E.P. Rathgeb, On the fairness of transport protocols in a multi-path environment, Proceeding of IEEE ICC, 2012.
[11] M. Tortelli, I. Cianci, L.A. Grieco, G. Boggia, P. Camarda, A fairness analysis of content centric networks, Proceeding of IEEE International Conference on the Network of the Future (NOF), 2011.
[12] Y. Ren, H. Tang, J. Li, H. Qian, Transport protocols for fast long distance networks, J. Softw. 21 (7) (2010) 1576–1588.
[13] G. Wang, Y. Wu, K. Dou, Y. Ren, J. Li, AppTCP: the design and evaluation of application-based TCP for e-VLBI in fast long distance networks, Future Gen. Comput. Syst. 39 (2014) 67–74.
[14] G. Wang, Y. Ren, J. Li, Performance evaluation of TCP congestion control algorithms in fast long distance network, J. Commun. (4) (2014) 81–90.
[15] R.S. Cheng, D.J. Deng, Congestion control with dynamic threshold adaptation and cross-layer response for TCP Vegas over IEEE 802.11 wireless networks, Int. J. Commun. Syst. 27 (11) (2014) 2918–2930.
[16] J. Domżał, R. Wójcik, P. Chołda, R. Stankiewicz, A. Jajszczyk, Efficient congestion control mechanism for flow-aware networks, Int. J. Commun. Syst. 29 (4) (2016) 787–800.
[17] Y. Ren, Y. Zhao, P. Liu, K. Dou, J. Li, A survey on TCP incast in data center networks, Int. J. Commun. Syst. 27 (8) (2014) 1160–1172.
[18] S. Braun, M. Monti, M. Sifalakis, C. Tschudin, An empirical study of receiver-based aimd flow-control strategies for ccn, Proceeding of IEEE ICCCN, 2013.
[19] G. Carofiglio, M. Gallo, L. Muscariello, ICP: design and evaluation of an interest control protocol for content-centric networking, Proceeding of IEEE INFOCOM Workshop on Emerging Design Choices In Name Oriented Networking (INFOCOM NOMEN), 2012.
[20] S. Salsano, A. Detti, M. Cancellieri, M. Pomposini, N. Blefari-Melazzi, Receiver–driven interest control protocol for content-centric networks, Proceeding of ACM SIGCOMM Workshop on Information Centric Networking (ICN), 2012.
[21] S. Arianfar, P. Nikander, L. Eggert, J. Ott, Contug: a receiver driven transport protocol for content-centric networks, Proceeding of IEEE ICNP, 2010.
[22] G. Carofiglio, M. Gallo, L. Muscariello, M. Papalini, Multipath congestion control in content-centric networks, Proceeding of IEEE INFOCOM 2013 Workshop on Emerging Design Choices in Name-Oriented Networking, 2013.
[23] C. Jin, D.X. Wei, S.H. Low, FAST TCP: motivation, architecture, algorithms, performance, IEEE/ACM Trans. Netw. 14 (6) (2006) 1246–1259.
[24] L. Zhang, Why tcp timers don't work well, ACM SIGCOMM Comput. Commun. Rev. 16 (3) (1986) 397–405.
[25] I. Psaras, V. Tsaoussidis, Why tcp timers (still) don't work well, Comput. Netw. 51 (8) (2007) 2033–2048.
[26] M. Alizadeh, A. Greenberg, D.A. Maltz,J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, M. Sridharan, Data Center TCP (DCTCP), Proceeding of ACM SIGCOMM, 2010.
[27] K. Ramakrishnan, S. Floyd, A Proposal to add Explicit Congestion Notification (ECN) to IP, RFC2481, 1999.
[28] K. Ramakrishnan, S. Floyd, D. Black, The Addition of Explicit Congestion Notification (ECN) to IP, RFC3168, 2001.
[29] C. Yi, A. Afanasyev, I. Moiseenko, L. Wang, B. Zhang, L. Zhang, A case for stateful forwarding plane, Comput. Commun. Inf. Centric Netw. 36 (7) (2013) 779–791.
[30] T. Fu, Y. Li, T. Lin, H. Tan, H. Tang, S. Ci, An effective congestion control scheme in content-centric networking, Proceeding of 13th International Conference on Parallel and Distributed Computing, Applications and Technologies, 2012.
[31] S. Floyd, V. Jacobson, Random early detection gateways for congestion avoidance, IEEE/ACM Trans. Netw. 1 (4) (1993) 397–413.
[32] J. Padhye, V. Firoiu, D. Towsley, J. Kurose, Modeling tcp throughput: a simple model and its empirical validation, Proceeding of the ACM SIGCOMM '98, 1998.
[33] C. Paasch, O. Bonaventure, Multipath TCP, ACM Queue 12 (2) (2014) 1–12.
[34] Y. Ren, J. Li, S. Shi, L. Li, X. Chang, An interest control protocol for named data networking based on explicit feedback, Proceeding of the 11th ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS '15), 2015.
[35] M. Hovaidi Ardestani, Congestion Control in Information Centric Networking using Neural Networks Master Thesis, Aalto University, Finland, 2014.
[36] P. Bazmi, M. Keshtgary, A neural network based congestion control algorithm for content-centric networks, J. Adv. Comput. Sci. Technol. 3 (2) (2014) 214–220.
[37] N. Rozhnova, S. Fdida, An effective hop-by-hop interest shaping mechanism for CCN communications, Proceeding of IEEE INFOCOM NOMEN Workshop, 2012.
[38] Y. Wang, N. Rozhnova, A. Narayanan, D. Oran, I. Rhee, An improved hop-by-hop interest shaper for congestion control in named data networking, Proceeding of ICN'13, 2013.
[39] G. Carofiglio, M. Gallo, L. Muscariello, Joint hop-by-hop and receiver-driven interest control protocol for content-centric networks, Proceeding of ACM SIGCOMM Workshop on Information Centric Networking (ICN), 2012.
[40] F. Zhang, C. Xu, Y. Zhang, A. Reznik, H. Liu, C. Qian, A transport protocol for content-centric networking with explicit congestion control, Proceeding of the 23rd International Conference on Computer Communications and Networks (ICCCN), 2014.
[41] S. Athuraliya, V. Li, S. Low, Q. Yin, REM: active queue management, IEEE Netw. 15 (3) (2001) 48–53.
[42] G. Carofiglio, M. Gallo, L. Muscariello, M. Papalini, S. Wang, Optimal multipath congestion control and request forwarding in information-centric networks, Proceeding of IEEE ICNP, 2013.
[43] G. Carofiglio, M. Gallo, L. Muscariello, D. Perino, Modeling data transfer in content-centric networking, Proceeding of ITC23, 2011.
[44] G. Carofiglio, M. Gallo, L. Muscariello, Bandwidth and storage sharing performance in information centric networking, Proceeding of ACM SIGCOMM Workshop on ICN, 2011.
[45] G. Carofiglio, M. Gallo, L. Muscariello, On the performance of bandwidth and storage sharing in information-centric networks, Comput. Netw. (2013).
[46] G. Wang, T. Huang, J. Liu, J. Chen, Y. Liu, Modeling in-network caching and bandwidth sharing performance in information-centric networking, J. China Univ. Posts Telecommun. 20 (2) (2013) 99–105.
[47] A. Udugama, S. Palipana, C. Goerg, Analytical characterization of multi-path content delivery in content centric networks, Proceeding of IEEE Conference on In Future Internet Communications (CFIC), 2013.
[48] G. Zhang, Y. Li, T. Lin, Caching in information centric networking: a survey, Comput. Netw. 57 (16) (2013) 3128–3141.
[49] Y. Xu, Y. Li, et al., An adaptive per-application storage management scheme based on manifold learning in information centric networks, Future Gen. Comput. Syst. 36 (July 2014) 170–179.
[50] H. Xie, G. Shi, P. Wang, TECC: towards collaborative in-network caching guided by traffic engineering, Proceeding of IEEE INFOCOM Mini conference, 2012.
[51] M. Badov, A. Seetharam, J. Kurose, V. Firoiu, S. Nanda, Congestion-aware caching and search in information-centric networks, in: Proceeding of the 1st ACM International Conference on Information-Centric Networking, 2014, pp. 37–46.
[52] E. Yeh, T. Ho, Y. Cui, M. MBurd, R. Liu, D. Leong, VIP: a framework for joint dynamic forwarding and caching in named data networks, in: Proceeding of the 1st ACM International Conference on Information-Centric Networking, 2014, pp. 117–126.
[53] M. Amadeo, A. Molinaro, C. Campolo, M. Sifalakis, Transport layer design for named data wireless networking, The Proceeding of IEEE INFOCOM Workshop on Name-Oriented Mobility, 2014.
[54] M. Meisel, V. Pappas, L. Zhang, L. First, Broadcast later: topology-agnostic forwarding under high dynamics, Proceeding of ACITA, 2010.
[55] M. Amadeo, A. Molinaro, G. Ruggeri, E-CHANET: routing, forwarding and transport in information-centric multihop wireless networks, Comput. Commun. 36 (7) (2013) 792–803.
[56] F. Angius, G. Pau, M. Gerla, BLOOGO: BLOOm filter based GOssip algorithm for wireless NDN, Proceeding of ACM NoM, 2012.
[57] L. Wang, A. Afanasyev, R. Kuntz, R. Vuyyuru, R. Wakikawa, L. Zhang, Rapid traffic information dissemination using named data, Proceeding of ACM NoM, 2012.
[58] S. Lederer, D. Posch, C. Timmerer, C. Westphal, S.Liu, C. Mueller, A. Detti, D. Corujo, J. Wang, N. Murray, Adaptive Video Streaming over ICN, IRTF, ICNRG, Internet Draft, draft-irtf-icnrg-videostreaming-03.txt, Feb. 23, 2015.
[59] D. Kulinski, J. Burke, L. Zhang, Video streaming over named data networking, IEEE COMSOC MMTC E-Letter 8 (4) (2013) 6–10.
[60] A.J. Abu, B. Bensaou, J.M. Wang, Interest packets retransmission in lossy CCN networks and its impact on network performance, Proceeding of the ACM ICN'14, 2014.
[61] A. Kazi, H. Badr, Some observations on the performance of ccn-flooding, in: Proceeding of ICNC, 2014, pp. 334–340.
[62] M. Virgilio, G. Marchetto, R. Sisto, PIT overload analysis in content centric networks, in: Proceeding of the 3rd ACM SIGCOMM Workshop on ICN, 2013, pp. 67–72.
[63] S. Arianfar, P. Sarolahti, J. Ott, Deadline-based resource management for information-centric networks, Proceeding of the ICN'13, 2013.
[64] J. Chen, M. Arumaithurai, X. Fu, K.K. Ramakrishnanz, Reliable publish/subscribe in content-centric networks, Proceeding of the ICN'13, 2013.
[65] M. Amadeo, C. Campolo, A. Molinaro, Design and analysis of a transport-level solution for content-centric VANETs, Proceeding of IEEE ICC'13 - Workshop on Emerging Vehicular Networks: V2V/V2I and Railroad Communications, 2013.
[66] S. Oueslati, J. Roberts, N. Sbihi, Flow-aware traffic control for a content-centric network, Proceeding of IEEE INFOCOM Conference, 2012.
[67] M. Tortelli, I. Cianci, L.A. Grieco, G. Boggia, P. Camarda, A Fairness, Analysis of content centric networks, Proceeding of 2011 International Conference on the Network of the Future, 2011.

[68] S. Braun, M. Monti, M. Sifalakis, C. Tschudin, CCN & TCP coexistence in the future internet: should CCN be compatible to TCP? Proceeding of the 5th IFIP/IEEE Int'l Workshop on Management of the Future Internet, Symposium on Integrated Network Management, 2013.
[69] X. Jiang, J. Bi, Interest set mechanism to improve the transport of named data, ACM Comput. Commun. Rev. 43 (4) (2013) 515–516.
[70] D. Perino, M. Varvello, K.P.N. Puttaswamy, ICN-RE: redundancy elimination for information-centric networking, Proceeding of ICN'12 workshop, 2012.
[71] H. Yuan, P. Crowley, Experimental evaluation of content distribution with NDN and HTTP, Proceeding of INFOCOM, 2013.
[72] J. Zhou, Q. Wu, Z. Li, M.A. Kaafar, G. Xie, A proactive transport mechanism with explicit congestion notification for NDN, Proceeding of ICC, 2015.
[73] N. Rozhnova, S. Fdida, An extended hop-by-hop Interest shaping mechanism for content-centric networking, Proceeding of IEEE GLOBECOM, 2014.
[74] Y. Zhu, H. Eran, D. Firestone, C. Guo, M. Lipshteyn, Y. Liron, J. Padhye, S. Raindel, M.H. Yahia, M. Zhang, Congestion control for large-scale RDMA deployments, Proceeding of SIGCOMM'15,, 2015.
[75] A. Feldmann, Internet clean-slate design: what and why? ACM SIGCOMM Comput. Commun. Rev. 37 (3) (2007) 59–64.
[76] J. Rexford, C. Dovrolis, Future internet architecture: clean-slate versus evolutionary research, Commun. ACM 53 (9) (2010) 36–40.
[77] J. Pan, S. Paul, R. Jain, A survey of the research on future internet architectures, IEEE Commun. Mag. (2011).
[78] M.F. Bari, S. Chowdhury, R. Ahmed, R. Boutaba, B. Mathieu, A survey of naming and routing in information-centric networks, IEEE Commun. Mag. (2012).
[79] A. Hoque, S.O. Amin, A. Alyyan, B. Zhang, L. Zhang, L. Wang, NLSR: named-data link state routing protocol, Proceeding of ACM ICN'13, 2013.
[80] C. Yi, J. Abraham, A. Afanasyev, L. Wang, B. Zhang, L. Zhang, On the role of routing in named data networking, Proceeding of ACM ICN'14, 2014.
[81] H. Yuan, T. Song, P. Crowley, Scalable NDN forwarding: concepts, issues and principles, Proceeding of the IEEE ICCCN, 2012.
[82] B. Hamdane, A. Serhrouchni, A. Fadlallah, S.G.E. Fatmi, Named-data security scheme for named data networking, Proceeding of the 3rd International Conference on the Network of the Future (NOF), 2012.
[83] K. Lei, C. Hou, L. Li, K. Xu, A RCP-based congestion control protocol in named data networking, Proceeding of IEEE International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2015.
[84] A. Karami, ACCPndn: adaptive congestion control protocol in named data networking by learning capacities using optimized time-lagged feedforward neural network, J. Netw. Comput. Appl. 56 (1) (2015) 1–18.
[85] F. Zhang, Y. Zhang, A. Reznik, H. Liuc, C. Qiand, C. Xue, Providing explicit congestion control and multi-homing support for content-centric networking transport, Comput. Commun. 69 (2015) 69–78.
[86] T. Nguyen, B. Ha, T. Le, P. Vo, Coupled multipath congestion control at receiver in content-centric networking, Proceeding of 2nd National Foundation for Science and Technology Development Conference on Information and Computer Science (NICS), 2015.
[87] D. Nguyen, M. Fukushima, K. Sugiyama, A. Tagami, Efficient multipath forwarding and congestion control without route-labeling in CCN, Proceeding of 2015 IEEE International Conference on Communication Workshop (ICCW), 2015.
[88] C. Zhang, L .Wang, Z. Wang, S. Wang, A service-based congestion control strategy for content centric networking, Proceeding of International Symposium on Computers & Informatics (ISCI), 2015.
[89] Y. Yu, A. Afanasyev, L. Zhang, Technical Report NDN-0034, 2015.
[90] A. Azgin, R. Ravindran, G. Wang, Mobility study for named data networking in wireless access networks, Proceeding of ICC, 2014.
[91] D.P. Arjunwadkar, Introduction of NDN with comparison to current internet architecture based on TCP/IP, Int. J. Comput. Appl. 105 (5) (2014) 31–35.
[92] D. Thaler, C. Hopps, Multipath issues in unicast and multicast next-hop selection, RFC 2991 (2000).
[93] A. Gurtov, T. Polishchuk, Secure multipath transport for legacy Internet applications, Proceeding of the 6th IEEE International Conference on Broadband Communications, Networks, and Systems, 2009.
[94] Y. Gu, RobertL. Grossman, UDT: UDP-based data transfer for high-speed wide area networks, Comput. Netw. 51 (7) (2007) 1777–1799.
[95] Y. Ren, H. Tang, J. Li, and H. Qian, A novel congestion control algorithm for high performance bulk data transfer. In Proceeding of the 8th IEEE International Symposium on Network Computing and Applications (NCA'09), 2009
[96] R. Hamilton, J. Iyengar, I. Swett, A. Wilk, QUIC: A UDP-Based Secure and Reliable Transport for HTTP/2, IETF, draft-tsvwg-quic-protocol-02, 2016.
[97] W. Richard Stevens, TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms, RFC 2001 (1997).
[98] G. Marfia, C. Palazzi, et al., TCP Libra: Exploring RTT-Fairness for TCP, NETWORKING 2007, LNCS 4479 (2007) 1005–1013.
[99] Y. Ren, J. Li, S. Shi, L. Li, G. Wang, An explicit congestion control algorithm for named data networking, Proceeding of the 35th IEEE INFOCOM Workshop on Multimedia Streaming in Infomation-/Content-Centric Networks (INFOCOM MuSIC), 2016.