一般社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
CAS2019-108,CS2019-108(2020-02)

Diffusion Type Packet Flow Control in Named Data Network

Chenglin LEE† and Hidenori NAKAZATO†

† Department of Computer Science and Communications Engineering, Waseda University
Okubo 3-4-1, Shinjuku-ku, Tokyo 169-8555, Japan

Abstract Named Data Networking (NDN) is a new network architecture. One of the feature of NDN is that routers may cache the content when data packets arrived. End-to-end methods used by traditional network cannot be applied in NDN. Therefore, researchers proposed hop-by-hop method, which is motivated by thermal diffusion. At present, there has been research about interest packets' flow control in diffusion method. However, the flow control for data packets have not been studied. In this paper, we propose a method to improve the performance of data packets flow control and verify it.

Key words Network, Named Data Network, Flow control, Diffusion

1. Introduction

The Internet we are using nowadays is constructed by many techniques such as TCP, IP, and so on. In traditional networks, users use end-to-end method to communicate with each other. In other words, they use the location information to know where to send and receive data. However, with the increase of mobile equipment's, communication for contents such as video or music is more and more important than communication with specific users. Therefore, researchers proposed a new network architecture which is mainly oriented to contents, called Named Data Networking (NDN) [1] [2].

A major feature of NDN is that the routers in NDN will cache the content when data is arrived. When a user wants to access the content, the router on the way may have the data the user wants in the cache and send data to user instead of the content provider. Therefore, the end-to-end method used by traditional networks cannot be applied in NDN. For this problem, researchers proposed a method called hop-by-hop method.

One hop-by-hop method is motivated by thermal diffusion phenomenon. Thermal diffusion is that when there is a heat source, the heat will spread from the heat source to the lower temperature place. This diffusion is similar to packet flow between the routers. The router with packet congestion, which means that it has lots of packets in the router, is just like heat source in thermal diffusion. Other routers with no congestion, which mean packets in routers are low, is like low temperature place in thermal diffusion. We want to let packets transmit from routers with congestion to other routers, and thermal diffusion might be useful. Therefore, researchers begin to study how to use this hop-by-hop approach to achieve packet flow control in NDN.

At present, there has been research on applying diffusion method to interest packets in NDN. However, the flow control applying the diffusion method to data packets has not been studied. This research is to study how to improve the data packets flow exploiting the thermal diffusion method.

2. Named Data Network

2.1 Introduction

The basic concept of NDN is that every data content is assigned a name. It is different from traditional network's point-to-point communication using IP addresses. NDN aims to improve the traditional IP architecture, to make routers process packets based on data rather than communication with endpoints. More specifically, the service provided by the traditional IP network is to deliver packets to the specified address, while the network service provided by NDN is to obtain data according to the specified name. The "name" in NDN can be anything, such as an endpoint, a picture, a movie, a part of data in a book, and an instruction to turn on the light. It is expected that the simple change of this concept will enable the Internet to be better used in a wider range of applications besides the existing point-to-point communication. In the design of NDN, many important functions inspired from tradition

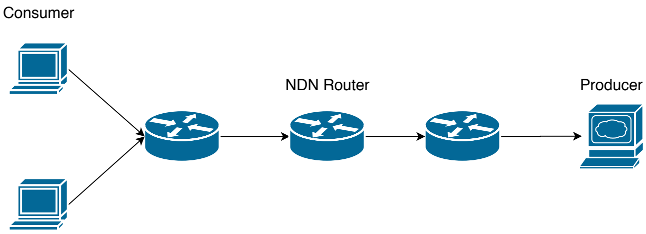

Fig. 1　Named Data Network

network have been integrated into the protocol from the beginning, such as using two different packets, interest packet and data packet, to self-regulate the network traffic and so on.

2. 2　NDN architecture

As shown in Figure 1, NDN consists of consumers who request data (content), producers that provide content for consumer, and routers that serve as relay. In NDN, consumers and producers use two different type packets to communicate with each other, which are interest packet and data packet. Interest packet is the packet that consumers put the name of the information they want and transmit it into the network. If the router that receives the interest packet has the requested data in the cache, it will return a data packet carrying the requested content to the consumer. If the content is not cached in the router, the interest packet will be forwarded to the next router. Data packet is the packet that is sent back when an interest packet arrives at a node, and the node has matching data. The data packet will be forwarded back the path of the corresponding interest packet in opposite direction.

In order to implement the forwarding function of interest packets and data packets, each router in NDN maintains three data structures, Pending Interest Table (PIT), Forwarding Information Base (FIB), and Content Store (CS).

Pending Interest Table (PIT):

Pending Interest Table stores information of interest packets forwarded by the router but not yet satisfied. Each PIT entry records the content name carried in the interest packet, as well as the interface where the the interest packet comes in and go out.

Forwarding Information Base (FIB):

In Forwarding Information Base, name prefix of the content and interface for transmitting an interest packet is stored in each entry. FIB is similar to the routing tables in IP routers, except that the number of faces to be sent is not necessarily only one. The router forwards the received interest packet to the face which content may exist according to its FIB.

Content Store (CS):

Content Store is used to cache data packets received by the router. If an interest packet requesting a content that matches to the content stored in the CS is received by router, the router will retrieves the content from its CS and returns a data packet containing the content to the consumer.

2. 3　Packet processing in NDN

At first, the consumer creates an interest packet containing the name of the content which the consumer requests and sends it to the network.

When the interest packet arrives at an NDN router, the NDN router will first query whether there is the correspond data in its CS. If there is correspond content, the router will directly return a data packet through the interface where the interest packet comes from, and then the interest packet will be discarded. Otherwise, the router will query the name of the interest packet in PIT. If there is the entry with the same content name in PIT, the router will simply record the source interface of the interest packet, but if there is no same entry in the PIT, the router will forward the interest packet to the next router based on the information in the FIB and the forwarding strategy of the router. If the router receives many interest packets with the same content name from the downstream nodes, it will only forward the first interest packet to the upstream producer.

With this action, if the interest packet finally reaches the producer, the producer will create a data packet containing the relevant content and sends it to the face where the interest packet has arrived.

When the data packet arrives at an NDN router, the NDN router will first query PIT and forward the data to all the downstream interfaces listed in the matched PIT entry, then remove the entry in the PIT and cache the data into CS. If the content name of the received data packet is not in the PIT, the router will discard the data packet.

As described above, the data packet is delivered to the consumer by tracing the arrival face that the internet packet left in the PIT.

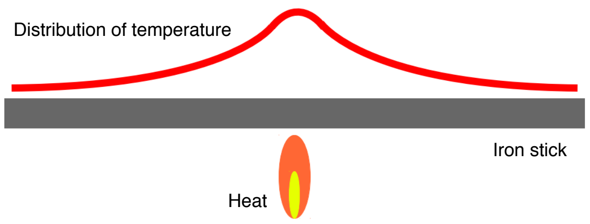

Fig. 2 Concept of thermal diffusion

3. Diffusion type packet control

3.1 Packet flow control in NDN

According to NDN architecture explained in the previous section, content which consumer requests might be sent from a router instead of its producer. Therefore, NDN is not suitable for traditional network communication using end-to-end method. On the other hand, the packet communication in the NDN is performed by the autonomous operation of each router. So, a hop-by-hop packet flow control might be more suitable for NDN. Diffusion type flow control has been proposed as one of the existing researches of autonomous packet flow control [3] [4]. Diffusion type flow control is a control system that packets in the whole network will work as diffusion to the desired direction. Specifically, it aims to avoid falling into congestion or to recover from the congestion state.

3.2 Introduction to diffusion type flow control

Diffusion type flow control is inspired from thermal diffusion. An example of thermal diffusion is shown in Figure 2.

Assume that a cold iron stick is prepared and the iron stick is heated for a moment. The heat diffuses as time elapses, and the temperature distribution across the iron stick is smoothed. If $n(x,t)$ at time t is defined as the heat quantity at position x on the iron stick at time t, the amount of heat in a micro interval $(x, x+dx]$ on the iron stick is $n(x,t)dx$. In the thermal diffusion, the heat flow in the iron stick $J(x,t)$ is determined by the difference in heat quantity between the adjacent micro intervals. That is

$$J(x,t) = -k\frac{n(x+dx,t) - n(x,t)}{dx} \tag{1}$$

In the form of partial differential is

$$J(x,t) = -k\frac{\partial n(x,t)}{\partial x} \tag{2}$$

Assume that the heat does not run outside the iron stick, and the heat does not flow from the outside into the iron stick except at the beginning, the temperature change in each time interval of the iron stick follows the result of the heat flow, and the following equation is established.

$$\frac{\partial n(x,t)}{\partial t} = -k\frac{\partial J(x,t)}{\partial x} \tag{3}$$

When expression (2) is substituted into formula (3), the following equation is obtained.

$$\frac{\partial n(x,t)}{\partial x} = k\frac{\partial^2 J(x,t)}{\partial x^2} \tag{4}$$

This equation is generally well known thermal diffusion equation.

3.3 Apply diffusion into network flow control

The behavior of such an iron stick is considered to correspond to the autonomous packet flow control system. An interval of the iron stick is considered to a router, and the heat in the interval is considered to the number of packets in the router in the NDN. Therefore, the action that the heat spread from the heat source to the lower temperature place can be considered as the packet flow control.

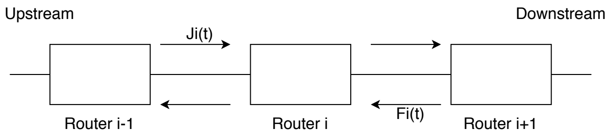

Fig. 3 Concept of thermal diffusion

As shown in Figure 3, a network with router $i (i = 1, 2, ..., N)$ connected in a straight line is considered, and the velocity between router i and downstream router $i+1$ is d_i. At the time t, the packet sending rate $J_i(t)$ from the router i to the downstream router $i+1$ is determined based on the local information that can be known by itself, and the packet is transmitted. Furthermore, the information of the router i is fed back to the upstream router $i-1$. Feedback information from the router i is named as $F_i(t)$. The packet sending rate $J_i(t)$ at router i to the downstream router $i+1$ is

$$J_i = \max(0, \min(L_i(t), J_i')) \tag{5}$$

$$J_i'(t) = r_i(t-d_i) - D_i(n_{i+1}(t-d_i) - n_i(t)) \tag{6}$$

In the equation, $n_i(t)$ is the number of packets in the router i at time t, $r_i(t-d_i)$ and $n_{i+1}(t-d_i)$ are included in the feedback information $F_{i+1}(t-d_i)$ which transmitted from the downstream router $i+1$ with the propagation delay d_i. $r_i(t-d_i)$ is the packet sending rate of the downstream router $i+1$ received after the propagation delay d_i at time t, and $n_{i+1}(t-d_i)$ is the number of packets at router $i+1$. $L_i(t)$ is the available bandwidth from router i to downstream router $i+1$. D_i is the diffusion coefficient.

Although there has been some researches [5] [6] [7] about diffusion type packet flow control so far, they are mainly focus on interest packet and its sending rate. Therefore, in this paper, we focus on data packet and propose a method to improve the performance of data packet flow control.

4. Data packet flow control method

4.1 Algorithm

In the previous study, only the interest packet distribution in the router was considered. In order to control data packet flow, we also include the number of current data packets in the router into calculation. That is, the formula to calculate the packet sending rate is modified as follows:

$$J_i = \max(0, \min(L_i(t), J_i')) \text{ and } \tag{7}$$

$$J_i'(t) = r_i(t-d_i) - D_i(n_{i+1}(t-d_i) - n_i(t) + p_i(t)) \tag{8}$$

where $p_i(t)$ is the number of data packet in the router i at time t. With this modification, we may be able to get better flow control.

4.2 Evaluation

In order to verify that the supposition is valid or not, we must use a simulator to simulate the results of the flow control. we used ndnSIM ver.1.0 [8], which is a representative Named Data Networking simulator based on NS-3, to make the experiment. In this paper, we only consider linear type network topology first. We make the simulation using 2 topology totally: the situation of one consumer to one producer, and the situation of multiple consumers to one producer. In the simulation, the packet size of the interest packet is 50 bytes and the packet size of the data packet is 1200 bytes. Moreover, all the simulation is simulated for 10 seconds. The diffusion coefficient D_i is 0.1. During the simulation, We will record number of packets sent by each router every 0.1 seconds and calculate the sending rate.

4.2.1 Topology 1: one consumer to one producer

The setup of the first topology is shown in Figure 4. There are one consumer (Src1) and one producer (Dst1),

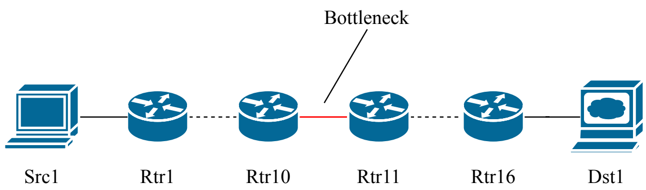

Fig. 4 Topology 1

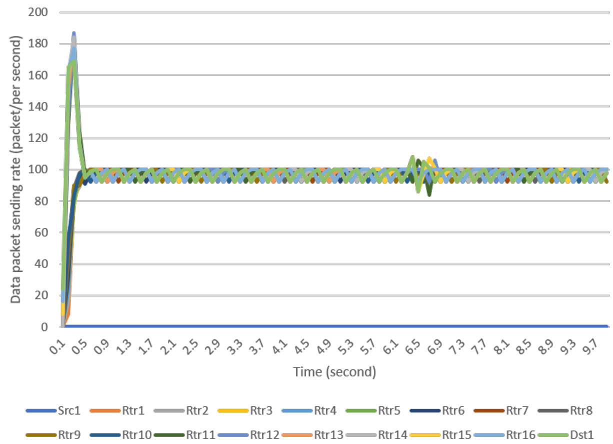

Fig. 5 data packet sending rate in topology 1

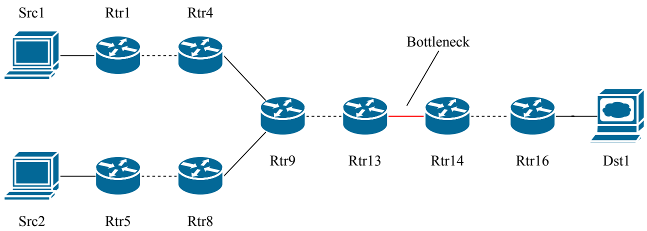

Fig. 6 Topology 2

and there are 16 routers (Rtr1 to 16) connected between the consumer and the producer. The link propagation delay between nodes is 5 ms, and Src1 transmits packets to Dst1 with 200 packets per second at the beginning. Between Rtr10 and Rtr11 there are a bottleneck link with 1 Mbps bandwidth, and the bandwidth of the other links in this topology is 10 Mbps.

The measurement result of topology 1 is as shown in Figure 5. According to Figure 5, we can see that at the beginning, data packets sending rate increase very fast. Then the diffusion method works, and we can see that data packet sending rate in all the routers and producer are close to 100, and become stable. Therefore, we can get the conclusion that the experiment is successful and the method is valid in this topology.

4.2.2 Topology 2: multiple consumers to one producer

The setup of the second topology is as shown in Figure 6. Rather than previous one, there are two consumer (Src1 and Src2) in this topology. Src1 and Src2 connect

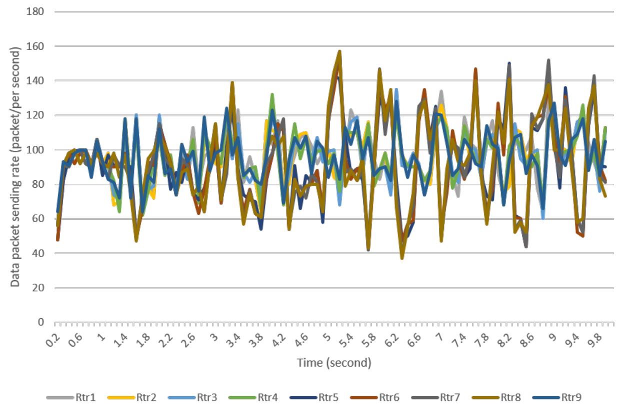

Fig. 7 data packet sending rate of Rtr1 to 9 in topology 2

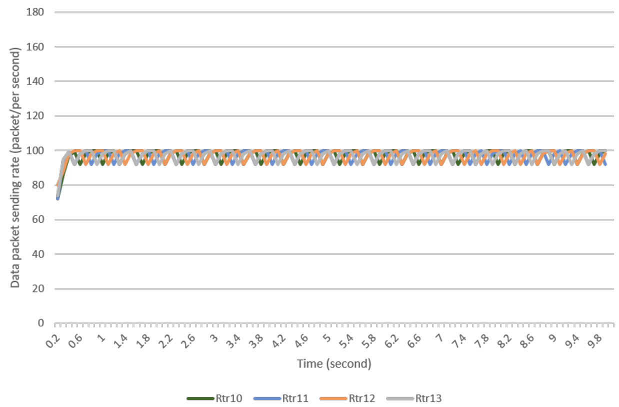

Fig. 8 data packet sending rate of Rtr10 to 13 in topology 2

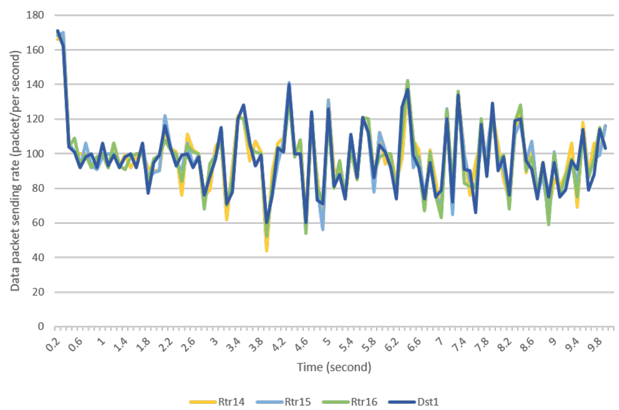

Fig. 9 data packet sending rate of Rtr14 to 16 and Dst1 in topology 2

to routers respectively, and merge at Rtr9. On the other hand, the producer is still one (Dst1), and connect to Rtr16. Bottleneck link is set between Rtr13 and Rtr14 with 1 Mbps bandwidth, and the bandwidth of the other links in this topology is 10 Mbps. Src1 and Src2 both transmits packets to dst1 with 200 packets per second at the beginning.

The measurement result of topology 2 is as shown in Figures 7, 8, and 9. According to the result, routers can be mainly classified into three parts.

The first part is the routers connected from the two sources to the router which merge happens, that is Rtr1 to Rtr9. At first, it temporarily become stable at the rate of 100, but soon it begins to shake up and down.

The second part consists of Rtr10 to Rtr13. They are routers after the merge but before the bottleneck. In Figure 8, we can see that these routers are very stable, which is the result we expect.

The third part is the routers after bottleneck, which are Rtr14 to Rtr16, and also the producer. According to Figure 9, the routers have very high speed sending data packets at the beginning, and then turn to stable for a while. However, it becomes up and down then.

Through these three kinds of result, we can suppose that the diffusion method still partially works because all routers become stable at the beginning and routers in the middle are always stable. However, there might be some issue that causing some of the routers unstable.

5. Conclusion

In this paper, we suppose a method to try to improve the performance of data packet flow control in NDN. Although the method we use gets good performance in topology 1, it might still have some issue should be discussed in the more complicated topology. Besides, at present, we only measure the situation of linear type network, but network in the real world is not so simple at all, so we will also measure the performance of data packet with more different topology as the future works.

Reference

[1] Van Jacobson, Diana K Smetters, James D Thornton, Michael F Plass, Nicholas H Briggs, and Rebecca L Braynard. Networking named content. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 1–12, 2009.

[2] Lixia Zhang, Deborah Estrin, Jeffrey Burke, Van Jacobson, James D Thornton, Diana K Smetters, Beichuan Zhang, Gene Tsudik, Dan Massey, Christos Papadopoulos, et al. Named data networking (ndn) project. *Relatório Técnico NDN-0001, Xerox Palo Alto Research Center-PARC*, 157:158, 2010.

[3] Chisa Takano and Masaki Aida. Diffusion-type autonomous decentralized flow control for open networks. In *24th International Conference on Distributed Computing Systems Workshops, 2004. Proceedings.*, pages 642–647. IEEE, 2004.

[4] Yongmao Ren, Jun Li, Shanshan Shi, Lingling Li, Guodong Wang, and Beichuan Zhang. Congestion control in named data networking–a survey. *Computer Communications*, 86:1–11, 2016.

[5] Yaogong Wang, Natalya Rozhnova, Ashok Narayanan, David Oran, and Injong Rhee. An improved hop-by-hop interest shaper for congestion control in named data net-

working. *ACM SIGCOMM Computer Communication Review*, 43(4):55–60, 2013.

[6] Kota Kaihori. Diffusion type interest packet sending rate reduction in named data network. Master's thesis, Waseda University, 1 2018.

[7] Klaus Schneider, Beichuan Zhang, and Lotfi Benmohamed. Hop-by-hop multipath routing: Choosing the right nexthop set. *arXiv preprint arXiv:1906.10266*, 2019.

[8] Alexander Afanasyev, Ilya Moiseenko, Lixia Zhang, et al. ndnsim: Ndn simulator for ns-3. *University of California, Los Angeles, Tech. Rep*, 4, 2012.