ITC 4/46

Journal of Information Technology
and Control
Vol. 46 / No. 4 / 2017
pp. 605-618
DOI 10.5755/j01.itc.46.4.15680

A Game Theoretic Framework for Congestion Control in Named Data Networking

Received 2016/07/13 | Accepted after revision 2017/11/14

http://dx.doi.org/10.5755/j01.itc.46.4.15680

A Game Theoretic Framework for Congestion Control in Named Data Networking

Jinfa Yao, Baoqun Yin, Xiaobin Tan, Yizhao Bao

University of Science and Technology of China, Department of Automation, 230027, Hefei, China
e-mail: jfyao@mail.ustc.edu.cn

Corresponding author: jfyao@mail.ustc.edu.cn

As an essential building block of the Named Data Networking (NDN) architecture, congestion mechanism has been the focus of much attention. The unique features of NDN architecture bring forward distinct requirements for congestion control from its Internet Protocol (IP) counterpart. In this paper, we present a game theoretic framework for flow rate control in NDN based on the concept of Nash bargaining solution from cooperative game theory, and propose a distributed flow-aware hop-by-hop congestion control mechanism on a solid analytical basis. In addition, we developed a possible implementation in the ndnSIM simulator and performed extensive simulations and performance evaluations to study the behavior of our scheme. We have seen that the proposed congestion control mechanism performs as designed, and significantly enhances the network performance.

KEYWORDS: cooperative game theory, congestion control, Nash bargaining, named data networking.

1. Introduction

The growing cumbersome architecture of current Internet and its inefficiency in terms of adapting to current network usage patterns push the research communities to quest for a more elegant design for the future Internet. Recently, Named Data Networking (NDN) [23] has been proposed as a clean-slate networking paradigm, which aims to directly address the challenges that arise from the incompatibility between communication models by shifting the Internet away from a host-centric paradigm to a more content-centric one, where data is uniquely identified by a name for addressing and caching.

NDN architecture offers a number of attractive advantages in enhancing network performance, such as network load reduction, low dissemination latency, and energy efficiency. One of the key new features that distinguish NDN from current Internet Protocol (IP) networks is that NDN networks are able to self-regulate traffic flows without relying on transport protocols. As suggested by NDN project group, the transport layer in traditional IP networks is removed out of the NDN architecture. Instead, NDN adopts adaptive forwarding [21] as the native transmission control mechanism. Besides, content delivery in NDN is performed in a pull-based fashion. To receive data, the requester needs to send out the Interest packets to trigger the corresponding data packets delivery on the reverse path followed by the requests. Due to the strict one-to-one relationship between Interest packet and data packet, flow balance is inherently held hop by hop in NDN networks. However, this property only enforces the basic transmission principle and is far from perfect.

Without a doubt, congestion is a fundamental issue for any data delivery networks and has been widely studied within the context of IP networks in the past decades. However, for content-oriented architectures where in-network caching and multipath forwarding are pervasive, as in the case of NDN, the existing congestion control mechanisms cannot be directly suitable for the following reasons. Firstly, in contrast with the push-based communication model of TCP/IP, NDN architecture advocates the pull-based paradigm where communication is driven by user requests. The inherent flow balance mechanism allows one to prevent the congestion caused by data packets in the downstream direction through pacing the Interest packet transmission rate at the requester side. Hence, it is convinced that the receiver-based congestion control mechanism is a better choice for NDN than the provider-driven one. Secondly, due to scalability and deployability, end-to-end schemes received more attention than hop-by-hop mechanisms in the context of host- and connection-based networks. However, NDN's location-independent, name-based routing protocol and its characteristic of extensive content multi-homing make it difficult to identify the Interest packets belonging to a specific source-destination pair. In addition, the extensive computation and onerous per-flow state management required for implementing flow-aware mechanisms is considered to be too costly for today's routers. However, the stateful forwarding plane in NDN routers makes per-Interest forwarding state available at each hop, which provides native support for flow-based control. Finally, as content delivery in NDN operates at the level of data chunks, each content item is segmented into a sequence of data chunks. Then according to the one-to-one principle, i.e., each Interest packet requests a particular data chunk, multiple Interest packets need to be transmitted in order to request large-sized contents, which may lead to uplink congestion as well. Therefore, it is inappropriate to neglect the fraction of traffic caused by Interest packets especially in the case of bidirectional flows. From the discussions above, it is clear that hop-by-hop flow-aware Interest packet rate control mechanisms are natural candidates for congestion control in NDN architecture when we seek to realize the optimal data delivery performance. Note that, in this paper, we use the terminology "flow" to represent the stream of Interest packets and data packets that share the same object name regardless of the corresponding receivers and repositories. The granularity of the flow, however, depends on the length of the name prefix.

Although there is now a rapidly growing literature in the context of NDN, the problem of congestion control in NDN still remains open. In this paper, we focus on developing a practical algorithm that allows a NDN router to dynamically adapt the transmission rate of Interest packets to the available network resources, with the objectives of proportionally fair resource allocation among the concurrent flows and low degree of implementation complexity. To achieve these goals, we formulate a game theory-based optimization framework for the congestion control problem in NDN environment with the constraints imposed on both Interests and data traffic in both uplink and downlink directions. The rationale for considering the interdependence between Interests and contents is that both Interests and contents contribute to congestion [2, 20]. By decomposing the Lagrange function of the primal problem, we get the explicit expression of the dual problem. Then using the gradient projection method, the optimal solution of the dual problem is approached and a distributed, hop-by-hop, flow-aware congestion control algorithm is proposed, which has been shown to be efficient under a variety of network scenarios through simulations.

The paper is organized as follows. In Section 2 we briefly review the related work and classify the existing congestion control schemes in the context of

NDN. In Section 3 we first present the necessary notations and assumptions, and then construct our optimization problem for fair flow rate allocation. Furthermore, we show that the optimal solution can be drawn in a distributed manner. In Section 4 we discuss a possible implementation of the proposed mechanism and some crucial issues of other related system functions. The balance of the paper includes preliminary simulations of the proposed mechanism in Section 5, and concluding remarks in Section 6.

2. Related Work

In recent years, much research effort has been put into the NDN area. A fair amount of NDN congestion control schemes have been put forward in the literature. According to the role responsible for reacting to network congestion, these schemes can be broadly classified into three categories: receiver-based mechanism, hop-by-hop control mechanism and the integration of both. Readers are referred to [17] for a survey of these approaches.

As aforementioned, in the context of NDN, or more generally of a pull-based network, receiver-based mechanisms are obviously more appropriate than their sender-driven counterparts in terms of traffic control. In [5], the authors design a receiver-driven Interest Control Protocol(ICP), which regulates the Interest expression rate at the receivers in a TCP-like window-based Additive Increase Multiplicative Decrease (AIMD) fashion. The RAAQM scheme described in [6] is an extended version of ICP, which enables multipath control by using a per-route mechanism to establish distinct congestion windows for controlling the Interest rate of each flow. The CCTCP [19] is also a receiver-driven timeout-based mechanism, but the receiver uses a special mechanism, “Anticipated Interests”, to predict the location of content on the path and maintains a separate retransmission time out for each anticipated source. Different from above implicit congestion control methods, Explicit Control Protocol (ECP) [18] adopts an explicit congestion detection and notification method, where receivers adjust their Interest sending rate according to the explicit congestion information detected and fed back by the routers.

The limitation of receiver-based congestion control mechanisms lies in two aspects: time lag of endpoints loss detection and hardness of handling traffic burst correctly. As an alternative or in conjunction with receiver-driven mechanisms, hop-by-hop approaches provide a way to avoid congestion proactively and response to congestion effectively on time when congestion occurs. In [15], a hop-by-hop Interest shaping mechanism (HoBHIS) was first introduced for CCN congestion control, where Interest shaping rate is dynamically adjusted based on the instantaneous queue occupancy, available capacity for data packets, as well as the response delay. A further work has been carried out in [16] to take advantage of an explicit feedback mechanism to control the client behavior and prevent a potential risk of network congestion. A popularity-based control scheme [14] was designed for NDN, which reacts to congestion by sending Interests for the expected data to be retrieved in the near future based on predicting the content popularity. The neural network based congestion control methods [3, 8] utilize the designed neural network to predict adaptively the existence of the congestion on links given the status of the network and avoid it.

Being aware of the deformities of the above two kinds of mechanisms, some researchers argued that NDN congestion control should be implemented by considering both parts of the data consumers and the network side. In the recent paper [7], a hybrid mechanism, HR-ICP, is proposed, which couples the receiver-driven congestion control ICP with hop-by-hop Interest shaping, to realize faster congestion avoidance and guarantee efficient bandwidth sharing. A similar work presented in [13] introduces a traffic control framework taking into account the Interest forwarding strategies at the consumers and the routers based on per-flow fair bandwidth sharing. The proposal in [22] discusses a Chunk-switched Hop Pull Control Protocol (CHoPCoP), which uses a random early marking to explicitly signal network congestion, besides a per-hop fair share Interest shaping and a receiver-side Interest control method.

In practice, it is often the case that both Interests and contents competing for the same link resources in the context of NDN. However, an interesting observation is that the impact of Interest packets on network congestion is usually left out in almost all prior work. A notable exception is the work presented in [20], where the interdependence between Interests and contents in bidirectional flows was first considered. In light of this factor, the authors presented a practical hop-by-hop Interest shaping algorithm, which we shall hereafter refer to as HIS scheme, to achieve high link utilization without congestive data loss. However, the HIS scheme does not distinguish between flows. Although a scheme of this kind is simpler to implement than a scheme that requires monitoring and controlling traffic on a per flow basis, we believe that it is necessary to supplement NDN with flow-aware mechanisms in order to achieve optimal network resource utilization. The reason is that, in such architecture as the case of NDN where the sources are priori unknown and may vary over time subject to in-network cache dynamics and on-the-fly request forwarding mechanisms, the response delay that each flow suffers may vary largely, which would have significant effects on the network performance.

The work presented in this paper focuses on NDN congestion control, which is inspired by the work in [20], and differs in the following aspects. First, we cast the congestion control problem within a game theoretic framework. To the best of our knowledge, our work is the first attempt to tackle the congestion control problem by leveraging the game theory in the context of NDN. Second, we develop a hop-by-hop congestion control mechanism with the objective of proportionally fair resource allocation among concurrent flows by taking into account the flow characteristics. The rationale for this consideration is to provide a flexible mechanism to cater to heterogeneous applications. Third, our proposal brings an additional benefit of providing the concurrent flows with some protection against misbehaving receivers.

3. Problem Formalization

Table 1

Notations

Notation	Description
R_i	Router i $(i = 1, 2)$
ℓ_i	Links connecting a pair of routers
C_i	Capacity of link ℓ_i
$\mathcal{F}_\mathrm{i}$	The set of request flows triggered by router R_i between the router pair of interest
r_f^i	Interest arrival rate of flow f on router i
$x_f^I(x_f^D)$	Link Interest (data) rate of flow f
ρ_f^i	Average size ratio of data to Interest of flow f observed at router i
a_f	Average response time of flow f measured by the downstream router

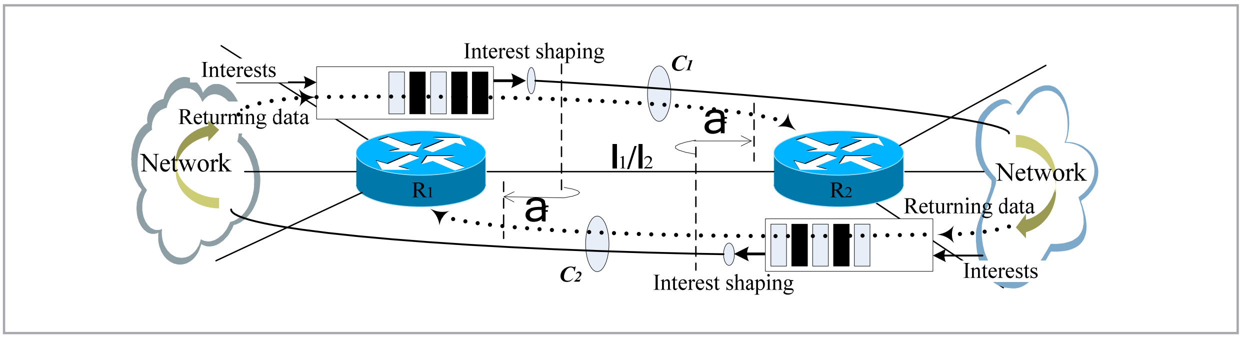
In this section, we first describe the network model and develop an optimization framework, then use this framework to explain decomposition. In order to aid the discussion in this section, we start with the introduction the notations used in this paper, which are summarized in Table 1.

3.1. The Model

We consider the traffic situation between a pair of NDN routers, denoted by R_1 and R_2, which are directly connected via two unidirectional links ℓ_1 and ℓ_2 in the opposite direction, as depicted in Figure 1. Each link ℓ_i is assumed to have a finite capacity, denoted by $C_i (i = 1, 2)$, which are static parameters. In addition, we define $\mathcal{F}_i$ the set of request flows (i.e. the Interest streams) triggered by router R_i between the router pair of interest. In our model, only the stream of Interest packets is shaped to pace their transmission, while the data packets are passed directly to the link

Figure 1

System model



output queue without shaping. We define r_f^i the Interest packet arrival rate of flow $f \in \mathcal{F}_i$ on the output interface corresponding to the link ℓ_i $(i = 1, 2)$ under steady state. The variables x_f^I and x_f^D denote the Interest shaping rate and data rate of flow *f*, respectively. The average size ratios between data packet and Interest packet of flow *f* in each direction are assumed to be ρ_f^1 and ρ_f^2. In addition, we introduce the variable a_f to represent the average instantaneous response delay suffered by the downstream router for retrieving the content *f* from the upstream one.

3.2. Optimization

To derive the optimization problem of the model described above, we make the common assumption that the utility, denoted by $Uf(\cdot)$, associated with flow *f* is an increasing, strictly concave, and continuously differentiable function of the Interest shaping rate x_f^I over the range $x_f^I > 0$ [9]. We further assume that the routers are individually maximizing the utilities of the flows in the request set it triggered. These utilities could mean different things to the routers, which we will discuss later. Another critical issue in network resources share is fairness, which has come to be seen as a required attribute to any well-designed congestion control mechanism. Although several definitions of fairness arise from various disciplines, in this paper, we consider the notion of proportional fairness [10]. In addition, from an implementation perspective, we particularly prefer that the functionalities of different routers be separated, with minimal communication overheads.

Based on above considerations, we borrow the concept of Nash bargaining [12] from cooperative game theory to define the optimal allocation of Interest shaping rate among flows. The reasons for this motivation are briefly summarized as follows. First, Nash bargaining scheme ensures that the joint system achieves a unique fixed operating point which takes both fairness and efficiency into account. Second, the Nash bargaining solution, obtained by maximizing the Nash product, is known to be Pareto optimal and proportionally fair from the economic theory viewpoint. Third, the Nash solution structure allows a modular implementation, which is indeed what we appreciate.

So far, we have discussed all the necessary basics of our model. We now define the flow rate allocation optimization problem for our NDN congestion control as follows, where the routers jointly solve the Nash product problem with linear constraints.

$$\text{maximize} \quad \prod_{f \in \mathcal{F}_1 \cup \mathcal{F}_2} U_f(x_f^I) \, . \tag{1}$$

$$\text{subject to} \quad \sum_{i \in \mathcal{F}_1} x_i^I + \sum_{k \in \mathcal{F}_2} x_k^D \leq C_1 \, . \tag{2}$$

$$\sum_{i \in \mathcal{F}_1} x_i^D + \sum_{k \in \mathcal{F}_2} x_k^I \leq C_2 \, . \tag{3}$$

$$0 \leqslant x_i^I \leqslant r_i^1, \forall i \in \mathcal{F}_1. \tag{4}$$

$$0 \leqslant x_k^I \leqslant r_k^2, \forall k \in \mathcal{F}_2. \tag{5}$$

where constraint (2) and (3) specify that the total flow rate in each direction is subject to the link capacity. Apparently, it is also required that the Interest shaping rate of a certain flow should not be more than its real demand and must be nonnegative. We characterize these constraints in (4) and (5). As the utility function $U_f(\cdot)$ is strictly concave and one can easily verify that the constraint set is convex, hence problem (1) is a convex optimization problem and has a unique solution. In what follows, we would look into how decomposition can be used to jointly optimize the Nash product problem with minimum communication overhead.

3.3. Decomposition and Solution

The idea of decomposition has been successfully used to solve large scale optimization problems and to solve separable problems in a decentralized manner [11]. Considering the objective of (1) which can be converted to maximize $\sum_{f \in \mathcal{F}_1 \cup \mathcal{F}_2} \log U_f(x_f^I)$, since the logarithmic function is monotonic, so the feasible solution space is unaffected. The introduction of the logarithmic functions helps reveal the decomposition structure of the original problem. In addition, according to the flow balance principle in NDN and above definitions, one can be readily to obtain following relationship, $x_f^D = \rho_f \cdot x_f^I$. Therefore, problem (1) can be rewritten as,

$$\text{maximize} \quad \sum_{f \in \mathcal{F}_1 \cup \mathcal{F}_2} \log U_f(x_f^I) \, . \tag{6}$$

$$\text{subject to} \quad \sum_{i \in \mathcal{F}_1} x_i^I + \sum_{k \in \mathcal{F}_2} \rho_k^2 x_k^I \leq C_1 \, . \tag{7}$$

$$\sum_{i\in\mathcal{F}_1}\rho_i^1 x_i^I + \sum_{k\in\mathcal{F}_2} x_k^I \le C_2 . \tag{8}$$

$$0 \leqslant x_i^I \leqslant r_i^1, \forall i \in \mathcal{F}_1. \tag{4}$$

$$0 \leqslant x_k^I \leqslant r_k^2, \forall k \in \mathcal{F}_2. \tag{5}$$

Then we take a partial Lagrangian of (6) with respect to the link capacity constraints as follows,

$$\begin{aligned}\mathcal{L}\left(x_f^I,\lambda,\mu\right) &= \sum_{f\in\mathcal{F}_1\cup\mathcal{F}_2}\log U_f(x_f^I) + \lambda\left(C_1 - \sum_{i\in\mathcal{F}_1}x_i^I - \right.\\ &\qquad \left.\sum_{k\in\mathcal{F}_2}\rho_k^2 x_k^I\right) + \mu\left(C_2 - \sum_{i\in\mathcal{F}_1}\rho_i^1 x_i^I - \sum_{k\in\mathcal{F}_2}x_k^I\right)\\ &= \sum_{i\in\mathcal{F}_1}\left\{\log U_i(x_i^I) - \left(\lambda + \mu\rho_i^1\right)x_i^I\right\} + \lambda C_1 + \\ &\qquad \sum_{k\in\mathcal{F}_2}\left\{\log U_k(x_k^I) - \left(\mu + \lambda\rho_k^2\right)x_k^I\right\} + \mu C_2\end{aligned} \tag{9}$$

where λ and μ are Lagrange multipliers associated with the link capacity constraints, which can be interpreted as the link price reflecting the cost of overshooting the link capacity. Observe that the Interest shaping rates x_f^I, $f\in\mathcal{F}_1$ ($\mathcal{F}_2$) can be separated in the Lagrangian function. Therefore, we take a dual decomposition approach, then problem (6) is decomposed into the following two subproblems: the optimization problem $G_{R_1}(\lambda,\mu)$, which is performed on router R_1, is given by (10),

$$\begin{aligned}G_{R_1}\left(\lambda,\mu\right) &= \max_{x_i^I(t)}\sum_{i\in\mathcal{F}_1}\left\{\log U_i(x_i^I) - \left(\lambda+\mu\rho_i^1\right)x_i^I\right\}\\ s.t. &\qquad 0 \le x_i^I \le r_i^1, \forall i \in \mathcal{F}_1.\end{aligned} \tag{10}$$

and the one performed on router R_2 is denoted as $G_{R_2}(\lambda,\mu)$:

$$\begin{aligned}G_{R_2}\left(\lambda,\mu\right) &= \max_{x_k^I(t)}\sum_{k\in\mathcal{F}_2}\left\{\log U_k(x_k^I) - \left(\mu+\lambda\rho_k^2\right)x_k^I\right\}\\ s.t. &\qquad 0 \le x_k^I \le r_k^2, \forall k \in \mathcal{F}_2.\end{aligned} \tag{11}$$

The optimal solutions of problem (10) and (11) for a given pair of parameters λ and μ define the dual function $D(\lambda, \mu)$. The dual problem is given as:

$$\begin{aligned}\min_{\lambda,\mu} D\left(\lambda,\mu\right) &= \sum_i G_{R_i}\left(\lambda,\mu\right) + \lambda C_1 + \mu C_2\\ \text{vars}. &\qquad \lambda \ge 0,\ \mu \ge 0.\end{aligned} \tag{12}$$

The dual problem described above can be easily solved based on a simple gradient projection method [4], which results that:

$$\frac{d}{dt}\lambda\left(t\right) = \alpha\left(t\right)\left(\sum_{i\in\mathcal{F}_1}x_i^I + \sum_{k\in\mathcal{F}_2}\rho_k^2 x_k^I - C_1\right) \tag{13}$$

$$\frac{d}{dt}\mu\left(t\right) = \beta\left(t\right)\left(\sum_{i\in\mathcal{F}_1}\rho_i^1 x_i^I + \sum_{k\in\mathcal{F}_2}x_k^I - C_2\right) \tag{14}$$

where $\alpha(t)$ and $\beta(t)$ are diminishing step sizes or small constant step sizes often used in practice, which will be discussed in Section 4. $\lambda(t)$ and $\mu(t)$ denote the implied cost at each iteration with $\lambda(0)$, $\mu(0)\in\mathcal{R}^+$ arbitrary, which can be taken to be 0.

Moreover, it is important to observe that the optimization subproblems performed on each router can be further solved in a distributed manner as well. That is, each flow maximizes the local Lagrangian $\mathcal{L}_f$:

$$\begin{aligned}\mathcal{L}_f\left(x_f^I\right) &= \log U_f(x_f^I) - \mathrm{I}_{\mathcal{F}_1}\left(f\right)\left(\lambda+\mu\rho_f^1\right)x_f^I\\ &\qquad - \mathrm{I}_{\mathcal{F}_2}\left(f\right)\left(\mu+\lambda\rho_f^2\right)x_f^I\end{aligned} \tag{15}$$

where $\mathrm{I}_y(x)$ is an indicated function, which is given by formula (16)

$$\mathrm{I}_y\left(x\right) = \begin{cases}1, & if\ x\in y\\ 0, & otherwise\end{cases}. \tag{16}$$

By differentiating (15), we have

$$\begin{aligned}\frac{d\mathcal{L}_f}{dx_f^I} &= \frac{U_f'(x_f^I)}{U_f(x_f^I)} - \mathrm{I}_{\mathcal{F}_1}\left(f\right)\left(\lambda+\mu\rho_f^1\right) - \mathrm{I}_{\mathcal{F}_2}\left(f\right)\left(\mu+\lambda\rho_f^2\right)\\ &= \frac{U_f'(x_f^I)}{U_f(x_f^I)} - H\left(f\right)\end{aligned} \tag{17}$$

where $H(f)$ is given by (18).

$$H\left(f\right) = \mathrm{I}_{\mathcal{F}_1}\left(f\right)\left(\lambda+\mu\rho_f^1\right) + \mathrm{I}_{\mathcal{F}_2}\left(f\right)\left(\mu+\lambda\rho_f^2\right). \tag{18}$$

Then by applying the gradient algorithm to solve the utility maximization subproblem in (15), we can draw that at each time t, $x_f^I(t)$ adapts according to (19).

$$\frac{d}{dt}x_f^I(t)=\gamma_f(t)\left[\frac{U_f'(x_f^I)}{U_f(x_f^I)}-H(f)\right] \quad (19)$$

where $\gamma_f(t)$ are sequences which define the step-sizes (or gains) associated with the iteration procedure.

4. Implementation and Deployment

In this section, we discuss some important issues related to the practical implementation of our proposed algorithm. We argue that this section only serves as a guideline and different implementations might be possible for other specific considerations.

4.1. Parameters Choices of the Master Problem

As described in the previous section, the parameter $\alpha(t)$ in formula (13) is a variable corresponding to link ℓ_1. For the sake of simplicity, in this paper, we assume that $\alpha(t)$ takes the following form, i.e., $\alpha(t) \equiv 1/C_1$, then the Lagrange multiplier $\lambda(t)$ can be interpreted as the propagation delay of link ℓ_1, as its evolution, i.e. the formula (13), is determined by a fluid queue evolution equation with input rate $\sum_{i\in\mathcal{F}_1}x_i^I+\sum_{k\in\mathcal{F}_2}\rho_k^2x_k^I$ and service rate C_1. Note that $\sum_{i\in\mathcal{F}_1}x_i^I$ and $\sum_{k\in\mathcal{F}_2}\rho_k^2x_k^I$ respectively denote the aggregate Interest stream and data stream that are incident on local output interface, where x_i^I $(i\in\mathcal{F}_1)$ is the solution of the optimization problem (10) so it can be derived locally. Moreover, both ρ_k^2 and $x_k^I(k\in\mathcal{F}_2)$ can be estimated locally as well by measuring the returning data sizes and the incoming Interest rate from the opposite direction. The same considerations are valid for the parameter $\beta(t)$, so we have that $\beta(t) \equiv 1/C_2$, and $\mu(t)$ can be seen as the link delay associated with link ℓ_2. Let $x_f^{I,\{\lambda(t),\mu(t)\}}$ $(f\in\mathcal{F}_1$ or $\mathcal{F}_2)$ be the optimal solutions to the problem $G_{R_i}(\lambda,\mu)$, $(i = 1, 2)$ for given $\lambda(t)$ and $u(t)$. Then, the updating iterations run at each router (i.e., R_1 and R_2) can be given by the following:

$$\lambda(t+1)=\left(\lambda(t)-\frac{1}{C_1}\left(C_1-\sum_{i\in\mathcal{F}_1}x_i^{I,\{\lambda(t),u(t)\}}(t)-\sum_{k\in\mathcal{F}_2}\rho_k^2x_k^{I,\{\lambda(t),u(t)\}}(t)\right)\right)^+ \quad (20)$$

$$\mu(t+1)=\left(\mu(t)-\frac{1}{C_2}\left(C_2-\sum_{i\in\mathcal{F}_1}\rho_i^1x_i^{I,\{\lambda(t),u(t)\}}(t)-\sum_{k\in\mathcal{F}_2}x_k^{I,\{\lambda(t),u(t)\}}(t)\right)\right)^+ \quad (21)$$

where $(x)^+\triangleq\max(x,0)$.

Moreover, in terms of information sharing, we append a novel tag that indicates the link information to the data packets. After each update, the router overrides the specified field in the data packets with the new value before forwarding the responding data. Notice that this is the only quantity which needs to be exchanged between the two routers in our model. Therefore, the communication overhead introduced by our approach is expected to be low.

4.2. Implementation of the Subproblems

In our decentralized model, each flow can optimize only its Interest shaping rate. One particular concern about our approach is the choice of the utility function $U_f(\cdot)$. Different alternatives of the utility function are possible as long as it satisfies the related assumptions in the previous section. By adopting different form of $U_f(\cdot)$, we end up with different controller. For instance, if we select $U_f(x_f^I(t))=K_1e^{-1/x_f^I(t)}/H^2(f)$ and $\gamma_f(t)=K_2(x_f^I(t))^2$, then we would derive a hop-by-hop window-based congestion control (i.e., a family of AIMD controllers) [6]. In this paper, for simplicity, we define the utility function of flow f as $U_f(x_f^I)=(x_f^I)^{w_f}$. Hence, for each flow, we can derive the following Interest transmission rate control:

$$x_f^I(t)=Min\left[r_f,\frac{w_f}{H(f)}\right] \quad (22)$$

where w_f denotes the weight of flow f. Alternative choices of weights $\{w_f\}$ may be possible. Notice that the network environments are characterized by a large quantity of heterogeneous applications whose demands and nature vary largely. The difference between the response times of each flow has a significant impact on fair sharing of resources. Moreover, customer satisfaction also plays a key role in quantifying the performance of congestion control mechanisms. Here, we aim at developing a mechanism which is able to tackle the response delay unfairness problem while taking into account the consumers' demands to cater to heterogeneous applications. With this in mind, we define the weight of each flow as

$w_f = a_f^{\delta} r_f^{1-\delta}$, where δ is a tunable factor. The parameter δ reflects how sensitive our proposed scheme is to the two different properties of flow. Hereafter, we simply assume that $\delta = 0.5$. According to our definition of w_f, it is required that each router needs to maintain an average response time estimation for each flow. To this end, we update the average estimated value of response time for each flow with exponential weighted moving average at every data packet reception, excluding retransmitted packets. Note that, in practice, $x_f^I(t)$ will be bounded by a finite constant (typically depended on the network conditions, such as the output link capacity, the number of active flows, and so on), which can provide the least protection against misbehaving consumers.

In addition, to implement such a flow-aware Interest shaping rate control, we need one virtual tail-drop queue per flow identified by the name of the content at every output interface. Each virtual queue controls the Interest transmission rate according to the optimal solution of the subproblem individually. As previously mentioned, such a scheme seems feasible in the NDN context, because the structure of NDN forwarding engine provides an inherent support to maintain the necessary information. Further, notice that the number of active flows depends on the flow granularity which can be handled flexibly in different scenarios. However, how to determine the flow granularity is a problem of the trade-off between scalability and performance, and we defer it to our future work.

Finally, we comment that our proposed congestion control mechanism is only a complement to the NDN architecture and should be coupled with other significant network functionalities, such as adaptive forwarding strategy, scheduling scheme, feedback mechanism, and caching policy, in order to achieve a satisfactory network performance. Moreover, although shaping mechanisms provide the network side with some degree of robustness against congestion, we argue that consumer strategies to react to congestion are also required to realize the maximum resource utilization objective. A sophisticated combination of these solutions might be appreciated but more comprehensive evaluations are needed. However, the details of all these above are beyond the scope of this paper. To simplify, in this paper, we assume that intermediate routers perform the best-route forwarding strategy. We further assume that Interests belonging to the same flow are forwarded equivalently rather than routed/forwarded individually. As for scheduling discipline, we adopt the simple First Come First Serve (FCFS) algorithm, which serves packets in order of arrival. In addition, we implement the same congestion feedback mechanism proposed in [21]. That is, once there is an Interest packet that cannot be forwarded because of being dropped by the shaper, a congestion signal NACK would be created and forwarded to the downstream. For the caching policies, we adopt the leave copy everywhere scheme as the caching placement policy and the least recently used strategy as the caching replacement policy.

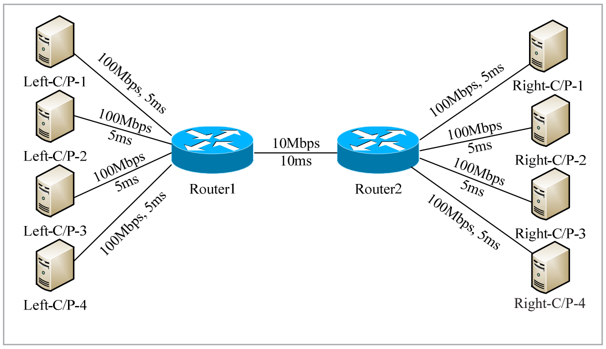
5. Performance Evaluation

In this section, we study the performance of the proposed mechanism using the ndnSIM [1] simulator. To this end, we carried out extensive simulations and comparisons with the methods in [5, 20] under a variety of network scenarios. In what follows, we present a selected set of simulation results to illustrate the properties and benefits of our scheme.

For what concerns the topology used for the evaluations, we consider the generic dumbbell topology shown in Figure 2, which is typically used for congestion control analysis. The rationales behind this are given as follows. First, in realistic networks, the data receiver and provider are generally connected by a path consisted of multiple links, however, to analyze congestion control mechanisms, one often ignores the existence of all the intermediate links, except for the bottleneck link that has the smallest bandwidth.

Figure 2

Network topology



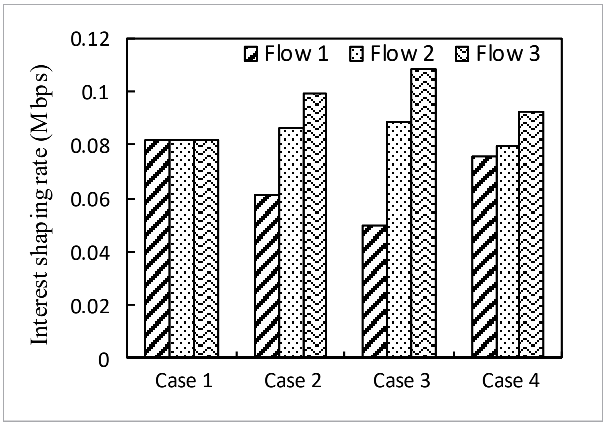
The dumbbell topology precisely provides such an approximation model for performance evaluation, which directly captures the bottleneck link. Second, this topology is believed to be, to some extent, representative of the behavior of a number of more general topologies, and allows us to explore the properties of our optimization model by manipulating the parameters (such as link delay, bandwidth, queue size, and so on). In fact, while the algorithms may exhibit slightly different behaviors in various topologies, some characteristics are common to many of them. Hence, a complete understanding of how the algorithms behave in one topology provides insight into how it might behave in other topologies. On these grounds, we are convinced that the simple dumbbell topology with a single bottleneck link is sufficient to validate the effectiveness and efficiency of our proposed method, and provides a natural starting point. In addition, all simulations assume the same average size ratio between contents and Interests, that is, a fixed content payload of 1000 bytes and a fixed size of 25 bytes for Interest packet. In addition, the packet buffer in each router is set at 100 packets. Simulations in every network scenario were performed multiple times to get its average performance.

5.1. Fairness Analysis

In the first scenario, we study the fairness property of our approach in the presence of unidirectional traffic. To this end, we launch three content retrievals, i.e., consumer Left-C/P-*i* (*i* = 1~3) retrieves the data with name prefix /Right-C-P-*i*/*k* from the corresponding producer on the right side, denoted by flow *i*. Consumers generate Interest packets at a constant rate. The start time of each flow is randomly determined between 0*s* and 1*s*. Unless otherwise specified, the topological parameters are set according to the initial configuration shown in Figure 2. By configuring flows with different response time and Interest expression rate, we observe the performance achieved by our algorithm under homogeneous and heterogeneous flow conditions. We are interested in the fair Interest shaping rate for each flow achieved by Router1 under the steady state. Simulation results under this scenario are reported in Figure 3.

Figure 3

Fairness evaluation under unidirectional traffic scenarios



As shown in Figure 3, here we consider four different cases. In the first case, we set all flows with the same response time and Interest expression rate, i.e., 40*ms* and 500*pkts/s*. As we can see, the total bandwidth of bottleneck link is equally shared among the competing flows as expected. We then vary either the response time or the Interest expression rate of each flow while keeping the other remaining unchanged as in the first case to investigate the effects of the two properties, respectively. In the second case, the link delays between Router2 and Right-C/P-1, Right-C/P-2, Right-C/P-3 are set to 5*ms*, 20*ms* and 35*ms*, respectively. In the third case, the Interest expression rates of each flow are 250*pkts/s*, 500*pkts/s* and 750*pkts/s*, respectively. As can be seen from the plot, our algorithm is able to converge to the theoretical optimal allowed rate suggested by formula (6) in both kinds of heterogeneous situations. In addition, it guarantees some degree of proportional fairness in terms of the throughput. In the last case, we vary the both properties of the flows. To specify, for the response time, the link delays of the top three edge links on the right side are 5*ms*, 15*ms* and 5*ms*, respectively. Moreover, flows 1~3 respectively issue Interest packets at a constant rate of 400*pkts/s*, 400 *pkts/s* and 600*pkts/s*. From the last two cases, we observe that our algorithm is in favor of "small" flows, which means that it gives priority to satisfy the demand of smaller flows.

In the second scenario, we consider bidirectional traffic and experimented with a slight variant of the above topology for fairness analysis. Here, we implemented data flows between each Left-/Right- C/P-*i* (*i* = 1~4) pair, among which flow 1 and 2 are launched by the Left-C/Ps while flow 3 and 4 travel in the op-

Table 2

Parameters setting

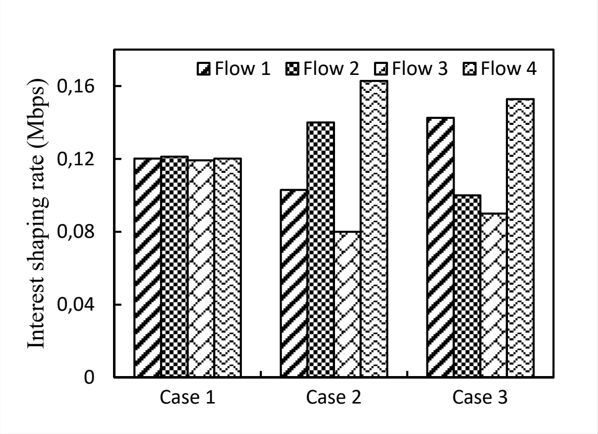
Flow	One-way delay				Interest expression rate ($pkts/s$)		
	Link name	Propagation delay (ms)					
		Case1	Case2	Case3	Case1	Case2	Case3
1	Router2 - Right-C/P-1	5	5	5	650	700	900
2	Router2 - Right-C/P-2	5	20	35	650	700	500
3	Router1 - Left-C/P-3	5	20	15	650	400	450
4	Router1 - Left-C/P-4	5	20	20	650	1000	850

posite direction. Each flow starts at a random time between 0 and 0.5s, and continues until the end of the evaluation. The specific parameters for the four flows in various cases are given in Table 2. Other parameters are the same as those in Figure 2. We compare the bargaining outcomes determined by the two routers under different cases.

Figure 4 illustrates the simulation results in this second simulation set. First, we observe that our scheme still works well in the presence of bidirectional traffic. As depicted in Figure 4, the same average behavior under both homogeneous and heterogeneous network settings shown in previous simulations of unidirectional flows is observed in this scenario, which can be theoretically proved based on the analysis in Section 3. The simulation results of Case2 also points out that our scheme is able to alleviate the potential unfairness caused by response delay or request rate.

Figure 4

Fairness evaluation under bidirectional traffic scenarios



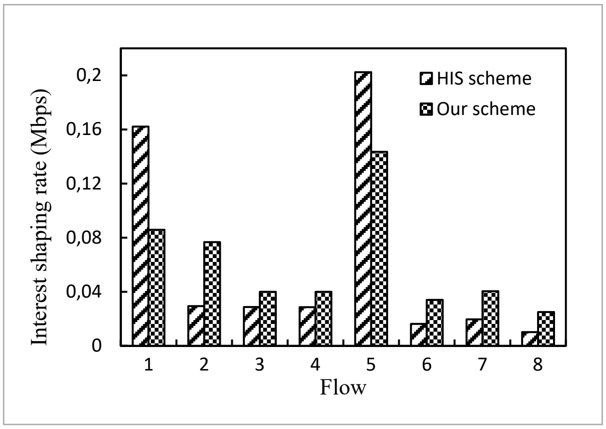
Note that by choosing an appropriate value for parameter δ, one may reach appreciative outcomes for specific objectives. However, here we defer the analysis about the design of parameter δ to our future work. Second, our shaping scheme achieves zero packet loss in all above cases, which improves the efficiency of data delivery by avoiding needless data transmission.

To explore the potentials of our proposed scheme, we run a third set of simulations and compare the results with the HIS scheme [20]. The same dumbbell topology is used as in the previous evaluations. In this test, we consider a symmetrical topology. The symmetry here means that, for each Left- and Right-C/P pair, the edge links on both sides have the same link parameters. We set the link delays between Router1 and Left-C/P-1, Left-C/P-2, Left-C/P-3, Left-C/P-4 to 5ms, 20ms, 15ms and 35ms, respectively. In addition, all edge links have a bandwidth of 100$Mbps$, and the parameters for the bottleneck link are propagation delay 10ms, bandwidth 10$Mbps$. Then we make the two parts of each Left-/Right-C/P-i ($i = 1$~4) pair request data from each other. The simulated permanent flows are as follows: consumer Left-C/P-i ($i = 1$~4) respectively issues Interest packets with a constant rate of 1000$pkts/s$, 400$pkts/s$, 200$pkts/s$ and 200$pkts/s$, denoted by flows 1~4 (also referred to as Group One), while the Interest expression rate of consumer Right-C/P-1 is 1000$pkts/s$ and the Right-C/P-2~4 adopt the smart AIMD strategy to adjust their Interest expression rate, denoted by flows 5~8 (Group Two). We compare the behaviors of both schemes under the situations with misbehaving flows and different consumer strategies. Figure 5 shows the comparison of the test results.

As we can see from the plot, both schemes achieve an

Figure 5

Fairness evaluation against misbehaving receivers



approximate full use of the bandwidth of bottleneck link in both directions. However, as HIS scheme imposes the same Interest rejection proportion on all concurrent flows in the same direction, hence the misbehaving consumer, i.e., flow 1, occupies most of the available link bandwidth in comparison with flows 2~4. This situation may become even worse when the honest consumers adopt the smart AIMD strategy but the misbehaving consumer still keeps requesting the contents greedily, as the results of flows in Group Two show. In contrast, we observe that our scheme provides a better control of consumer misbehavior. This is attributed to the fact that our scheme inclines to satisfy small flows preferentially. Accordingly, for flows in Group One set, the heavy flow is penalized and the honest consumers achieve almost the maximum throughput under our scheme. Even though there is some performance degradation in the second flow group due to effect of the consumer strategy, our scheme also outperforms the HIS scheme by providing fairness among competing flows in a more rational way.

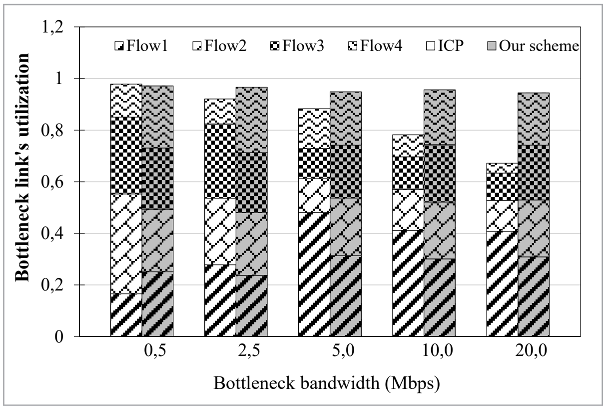
5.2. Bottleneck Link's Utilization

We then compare the proposed mechanism with ICP [5] in terms of the bottleneck link's utilization. ICP is a pure receiver-based end-to-end scheme, where receivers detect congestion via timeout mechanism and regulate Interest expression rate according to a TCP-like AIMD principle individually without any congestion control at intermediate routers or servers. In this scenario, we conducted the simulations on the symmetrical dumbbell topology under bidirectional traffic condition, with 5*ms*, 20*ms*, 20*ms* and 35*ms* link delay for the edge links on left side respectively, and 100*Mbps* for all edge links. We analyzed the performance of both mechanisms by varying the bandwidth of the bottleneck link. Figure 6 shows the selected results out of our extensive simulations, where we only presented the bottleneck link's utilization achieved in one of the directions, noting that a slight fluctuation at a range of up to 0.01 is applicable to the performance achieved in the other direction in all cases.

As can be seen from Figure 6, the ICP scheme suffers from severe performance degradation when we vary the bottleneck bandwidth from 0.5*Mbps* to 20 *Mbps*, whereas our proposed scheme is able to achieve almost full utilization of the bottleneck bandwidth in all cases. In addition, it is clear that the throughput ratio of concurrent flows under ICP scheme becomes more and more deformed as the bottleneck bandwidth increases, which results from the heterogeneous RTTs. However, as the figure shows, our scheme achieves rather good throughput fairness as expected even though in the presence of heterogeneous RTTs. This also confirms the fairness analysis that we presented in the previous subsection.

Figure 6

Bottleneck link's utilization



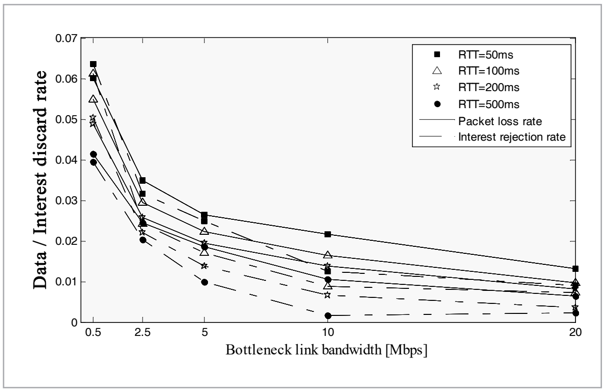
5.3. Data Loss or Interest Rejection

In this simulation set, we evaluate the data packet loss erformance under the two-way traffic scenario. To this end, we consider the same network set-up shown in Figure 2 and vary the bandwidth and latency of the bottleneck link to study their impacts on the performance. We are interested in the data packet loss rate or the Interest rejection rate on the bottleneck link. Figure 7 shows the performance comparison of our approach and ICP scheme.

As ICP infers congestion based on data packet loss, hence, we observe a non-ignorable data packet loss rate in all cases. Instead, noting that congestion is proactively prevented by early Interest rejection under our scheme, therefore, no data packet loss is observed on the bottleneck link, which also confirms the effectiveness of our scheme. Figure 7 shows how improvements in link bandwidth reduce data packet loss rates and Interest rejection rates. As can be seen from the plot, the benefits obtained from bandwidth improvements become smaller and smaller as bandwidth increases. In addition, the discard rate of data packet (or Interest) also decreases when we vary the bottleneck link latency so that the RTT ranges from 50*ms* to 500*ms*, which may be due to the throughput degradation under long RTT. Finally, we observe that the Interest rejection rates of our scheme is lower than the data packet loss rate of ICP except for the first case where the bottleneck bandwidth is 0.5*Mbps*, which is due to the fact that our scheme achieves a relatively higher link utilization.

Figure 7

Packet loss rate vs. Interest rejection rate

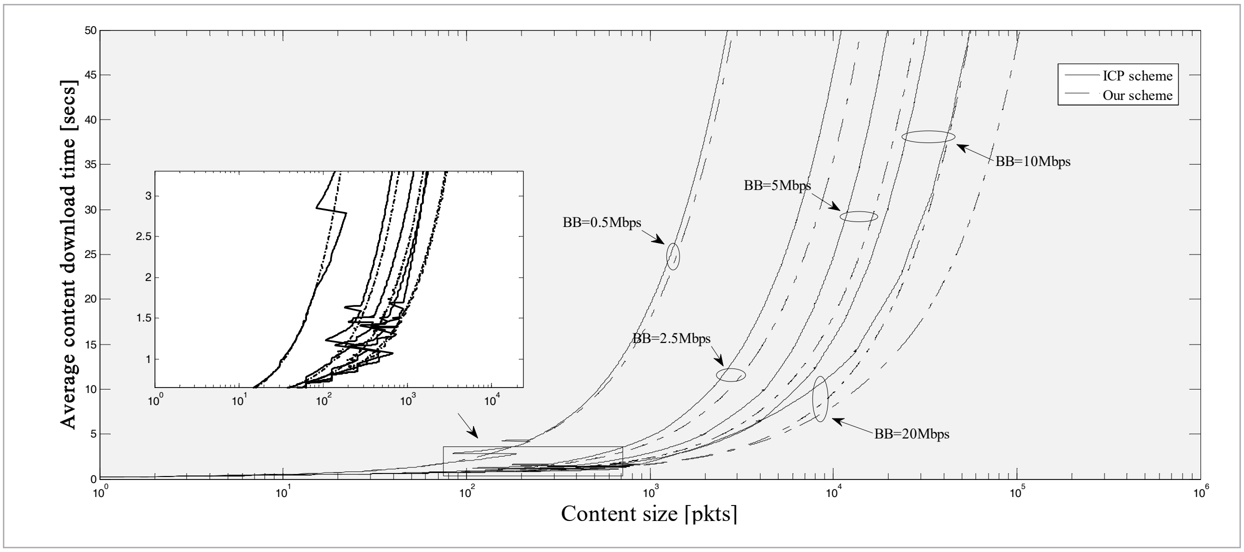


5.4. Content Download Time

We examine the download performance of our approach and the ICP scheme under the given network conditions in Figure 8. The simulated flows' RTTs were chosen to be a representative value of 100*ms*, and we presented the average results obtained from a number of simulation runs. As Figure 8 illustrates, our approach has a shorter content download time than ICP, regardless of the content size and the bottleneck bandwidth (BB). In particular, we observe that the performance gap between the two schemes become more and more significant as the bottleneck bandwidth increases. This is majorly attributed to the fact that our approach manages to achieve a higher throughput due to more efficient utilization of the bottleneck bandwidth relative to

Figure 8

Average content download time



ICP scheme. In addition, the fact that congestion is avoided by proactively dropping excessive Interest packets so that zero data packet loss rate is guaranteed by our scheme also partially explains the phenomenon (noting that the folds for the ICP curves in the graph are due to data packet loss). Since early Interest packet drop squanders fewer network resources than late data packet loss due to reduction of redundant content transmission, and consequently results in smaller download time.

6. Conclusion and Future Work

In the present paper, we discussed the specific requirements for congestion control mechanisms in NDN environment, and aimed at developing an effective hop-by-hop congestion mechanism that corresponds to the new features introduced in NDN architecture. With this in mind, we model the NDN congestion control problem as a flow rate allocation optimization problem based on the game theory. We have also proposed a distributed asynchronous algorithm to implement the solution. Simulations are conducted to validate the effectiveness and efficiency of our proposed approach, and the results demonstrate the benefits of the proposed hop-by-hop Interest rate control scheme in improving network performance, such as network throughput, resource utilization and fairness. Comprehensive performance evaluations are planned in our future work, which include an extension of our work to achieve the global optimization among all output interfaces as well as exploratory investigations on the satisfactory sophisticated forwarding strategies to combine with our mechanism. We will also explore the complexity and scalability of our scheme.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under grant Nos. 61233003, 61673360, in part by Research Fund for the Doctoral Program of Higher Education of China under grant No. 20123402110029 and in part by Natural Science Research Program of the Anhui High Education Bureau of China under grant No. KJ2012A286.

References

1. Afanasyev, A., Moiseenko, I., Zhang, L. ndnSIM: NDN Simulator for NS-3. University of California Technical Report, 2012.
2. Byun, D., Lee, B. J., Park, Y. Adaptive Interest Adjustment in CCN Flow Control. 2014 IEEE Global Communications Conference (GLOBECOM), 2014, 1873-1877. https://doi.org/10.1109/GLOCOM.2014.7037081
3. Bazmi, P., Keshtgary, M., A Neural Network Based Congestion Control Algorithm for Content-Centric Networks. Journal of Advanced Computer Science and Technology, 2014, 3(2), 214-220. https://doi.org/10.14419/jacst.v3i2.3696
4. Bertsekas, D. P., Tsitsiklis, J. N. Parallel and Distributed Computation: Numerical Methods. Englewood Cliffs, NJ, Prentice Hall, 1989.
5. Carofiglio, G., Gallo, M., Muscariello, L. ICP: Design and Evaluation of an Interest Control Protocol for Content-Centric Networking. 2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2012, 304-309.
6. Carofiglio, G., Gallo, M., Muscariello, L., Papali, M. Multipath Congestion Control in Content-Centric Networks. 2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2013, 363-368.
7. Carofiglio, G., Gallo, M., Muscariello, L. Joint Hop-by-Hop and Receiver-Driven Interest Control Protocol for Content-Centric Networks. Proceedings of the second edition of the ICN Workshop on Information-Centric Networking, ACM, 2012, 37-42. https://doi.org/10.1145/2342488.2342497
8. Karami, A. ACCPndn: Adaptive Congestion Control Protocol in Named Data Networking by Learning Capacities Using Optimized Time-Lagged Feedforward Neural Network. Journal of Network and Computer Applications, 2015, 56, 1-18. https://doi.org/10.1016/j.jnca.2015.05.017
9. Kelly, F. P., Maulloo, A., Tan, D. Rate Control in Communication Networks: Shadow Prices, Proportional Fairness and Stability. Journal of the Operational Research Society, 1998, 49, 237-252. https://doi.org/10.1057/palgrave.jors.2600523
10. Kelly, F. Charging and Rate Control for Elastic Traffic. European Transactions on Telecommunications, 1997, 8, 33-37. https://doi.org/10.1002/ett.4460080106

11. Mitrovic, D., Ivanović, M., Geler, Z. Agent-Based Distributed Computing for Dynamic Networks. Information Technology and Control, 2014, 43(1), 88-97. https://doi.org/10.5755/j01.itc.43.1.4588
12. Nash, J. F. The Bargaining Problem. Econometrica: Journal of the Econometric Society, 1950, 18 (2), 155-162. https://doi.org/10.2307/1907266
13. Oueslati, S., Roberts, J., Sbihi, N. Flow Aware Traffic Control for a Content-Centric Network. 2012 Proceedings IEEE INFOCOM, 2012, 2417-2425.
14. Park, H., Jang, H., Kwon, T. Popularity-Based Congestion Control in Named Data Networking. 2014 Sixth International Conference on Ubiquitous and Future Networks (ICUFN), 2014, 166-171. https://doi.org/10.1109/ICUFN.2014.6876774
15. Rozhnova, N., Fdida, S. An Effective Hop-by-Hop Interest Shaping Mechanism for CCN Communications. 2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2012, 322-327.
16. Rozhnova, N., Fdida, S. An Extended Hop-by-Hop Interest Shaping Mechanism for Content Centric Networking. IEEE GLOBECOM 2014, Austin, Texas, United States, 2014. https://doi.org/10.1109/GLOCOM.2014.7389766
17. Ren, Y., Li, J., Shi, S., Li, L., Wang, G., Zhang, B. Congestion Control in Named Data Networking – A Survey. Computer Communications, 2016, 86, 1-11. https://doi.org/10.1016/j.comcom.2016.04.017
18. Ren, Y., Li, J., Shi, S., Li, L., Wang, G. An Explicit Congestion Control Algorithm for Named Data Networking. 2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2016, 294-299.
19. Saino, L., Cocora, C., Pavlou, G. CCTCP: A Scalable Receiver-Driven Congestion Control Protocol for Content Centric Networking. 2013 IEEE International Conference on Communications (ICC), 2013, 3775-3780. https://doi.org/10.1109/ICC.2013.6655143
20. Wang, Y., Rozhnova, N., Narayanan, A., Oran, D., Rhee, I. An Improved Hop-by-Hop Interest Shaper for Congestion Control in Named Data Networking. ACM SIGCOMM Computer Communication Review, 2013, 43(4), 55-60. https://doi.org/10.1145/2534169.2491233
21. Yi, C., Afanasyev, A., Wang, L., Zhang, B., Zhang, L. Adaptive Forwarding in Named Data Networking. ACM SIGCOMM Computer Communication Review, 2012, 42(3), 62-67. https://doi.org/10.1145/2317307.2317319
22. Zhang, F., Zhang, Y., Reznik, A., Liuc, H., Qi, C., Xue, C. Providing Explicit Congestion Control and Multi-Homing Support for Content-Centric Networking Transport. Computer Communications, 2015, 69, 69-78. https://doi.org/10.1016/j.comcom.2015.06.019
23. Zhang, L., Afanasyev, A., Burke, J., Jacobson, V., Claffy, K., Crowley, P., Papadopoulos, C., Wang, L., Zhang, B. Named Data Networking. ACM SIGCOMM Computer Communication Review, 2014, 44(3), 66-73. https://doi.org/10.1145/2656877.2656887

Summary / Santrauka

As an essential building block of the Named Data Networking (NDN) architecture, congestion mechanism has been the focus of much attention. The unique features of NDN architecture bring forward distinct requirements for congestion control from its Internet Protocol (IP) counterpart. In this paper, we present a game theoretic framework for flow rate control in NDN based on the concept of Nash bargaining solution from cooperative game theory, and propose a distributed flow-aware hop-by-hop congestion control mechanism on a solid analytical basis. In addition, we developed a possible implementation in the ndnSIM simulator and performed extensive simulations and performance evaluations to study the behavior of our scheme. We have seen that the proposed congestion control mechanism performs as designed, and significantly enhances the network performance.

Kaip esminis įvardintų duomenų tinklaveikos (NDN) architektūros statybinis blokas, perkrovos mechanizmas yra susilaukęs daug tyrėjų dėmesio. Iš savo IP dublikato, unikalūs NDN architektūros bruožai pateikia išskirtinius reikalavimus perkrovos kontrolei. Straipsnyje pristatoma žaidimo teorinė sistema, skirta srauto greičio reguliavimui NDN, sukurta remiantis Nash derybų sprendimo koncepcija iš bendradarbiaujančių žaidimų teorijos. Remiantis solidžiu analitiniu pagrindu, taip pat siūlomas paskirstytas, į srautą reaguojantis šuolinės perkrovos valdymo mechanizmas. Taip pat sukurtas galimas įgyvendinimas ndnSIM simuliatoriuje, atlikta daug išsamių simuliacijų ir veiklos analizių sukurtos schemos elgsenai ištirti. Siūlomas perkrovos valdymo mechanizmas veikia taip, kaip suprojektuotas, ir gerokai padidina tinklo našumą.