Exploring Rate-Based Congestion Control in NDN

Sichen Song
University of California, Los Angeles
Los Angeles, CA, USA
songsichen@cs.ucla.edu

Lixia Zhang
University of California, Los Angeles
Los Angeles, CA, USA
lixia@cs.ucla.edu

ABSTRACT

We report our work-in-progress in designing a rate-based congestion control for NDN networks. We first explain why rate-based approach represents a more promising direction compared to the existing window-based congestion control solutions, and then provide a sketch of our initial design, together with a few lessons we have learned and preliminary results in pursuing this new direction.

CCS CONCEPTS

• **Networks**;

KEYWORDS

Rate-based congestion control, Named Data Networking

ACM Reference Format:
Sichen Song and Lixia Zhang. 2021. Exploring Rate-Based Congestion Control in NDN. In *8th ACM Conference on Information-Centric Networking (ICN '21), September 22–24, 2021, Paris, France.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3460417.3483379

1 INTRODUCTION

In a network running the Named Data Network (NDN) [3] protocol stack, data consumers fetch contents by sending interest packets carrying content names. Data packets are delivered to requesting consumers following the reverse path of corresponding interests, and can be cached at routers along the way. NDN's stateful forwarding plane also enables routers to utilize multiple paths in interest forwarding.

NDN's new features bring challenges to network congestion control. Although congestion control is a well studied, well-understood area in TCP/IP networking, the existing solutions are centered around a fundamental concept of "pipe size", i.e. the bandwidth-delay product (BDP) between two communicating ends with the delay being the minimum delay without queueing. Senders can achieve high bandwidth utilization and low queuing delay by adjusting the congestion window to be close to the BDP. However, the BDP concept is no longer valid in NDN due to in-network caching and dynamic multipath forwarding since data retrievals are no longer between two specific endpoints along a single path.

In this poster, we present our preliminary investigation into a rate-based congestion control (R-CC) design. Instead of controlling the congestion window size, R-CC compares a consumer's interest sending rate with the data arrival rate to detect congestion, and adjusts the interest sending rate accordingly. We provide a problem statement in §2, and a sketch of the R-CC design in §3. We report our preliminary results in §4 and future work in §5.

ICN '21, September22–24,2021, Paris,France

ACM ISBN 978-1-4503-8460-5/21/09.
https://doi.org/10.1145/3460417.3483379

2 PROBLEM STATEMENT

Since most existing congestion control solutions are centered on the BDP concept, several NDN congestion control proposals aim to estimate BDP in the presence of in-network caches and multipath forwarding, by requesting each NDN data packet to carry its retrieval path information. Our effort takes a different direction by articulating the impact that in-network caches and multipath forwarding may have on network traffic, which an end consumer can measure.

Intuitively, in-network caching and dynamic multipath forwarding can lead to fluctuations of round-trip time and data arrival rate. Therefore, an effective NDN congestion control solution must be adaptive to the available bandwidth over all active paths (Requirement-1), and be insensitive to RTT variations (Requirement-2).

To meet Requirement-1, we propose comparing a consumer's interest sending rate with the data arrival rate to detect congestion and adjust the interest sending rate accordingly. Since network congestion always *slows down* data arrival rate, congestion detection should not be affected too much by data arrival rate fluctuation. To meet Requirement-2, accurate data arrival rate measurements must not rely on a good RTT estimation. We propose to use the maximum observed RTT, MRTT, as the wait time after sending rate adjustment before measuring the effect of that adjustment. The maximum RTT can be much longer than the necessary waiting time, but it helps minimize control fluctuations. Furthermore, although it takes an RTT to learn whether the data arrival rate increases according to an interest sending rate increase, data arrival rate decrease should trigger immediate adjustment action, no matter how much time has passed since the last adjustment, because no interest rate adjustment should lead to reduced throuput. Thus the drop can only be due to external factors that increased network load.

3 DESIGN

The basic idea of our rate-based congestion control algorithm can be described as the following: each round, a consumer keeps its interest sending rate for MRTT, so that the data requested at the beginning of this round comes back. Then, it takes a pre-defined "measure" time to measure the data arrival rate, and compare it to the interest sending rate. Congestion can be detected by thresholding their difference. If data arrival is much slower than interest sending, congestion is detected, and interest sending rate will multiplicatively decrease. If the two rates are the same, the interest sending rate will increase following a cubic curve similar to CUBIC

TCP [2]. Fig. 1 shows the expected behavior of consumer in a fixed bandwidth network without caching or multipath forwarding.

To detect congestion caused by external factors that may occur at any time, the consumer continuously measures data arrival rate and checks whether it dropped. Data arrival rate drop will also trigger a multiplicative decrease, as illustrated in fig. 2.

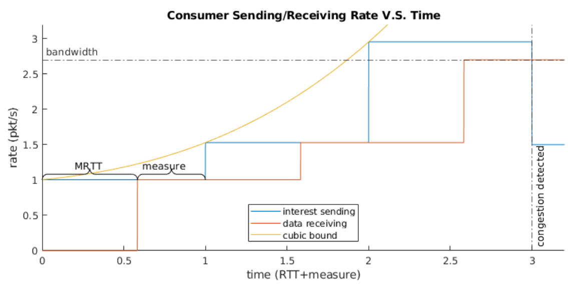

Figure 1: Consumer under fixed bandwidth

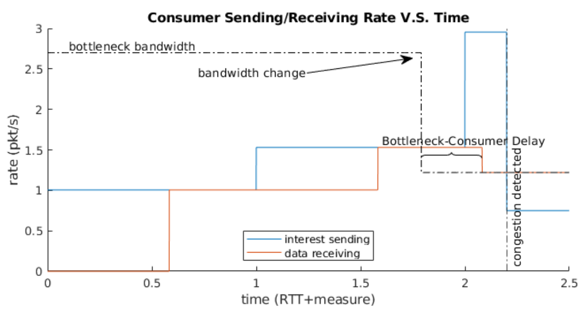

Figure 2: Consumer under bandwidth decrease

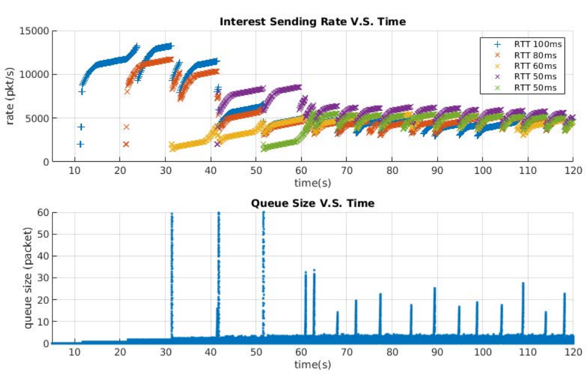

Figure 3: Five-flow dumbbell typology simulation

Below we discuss a few lessons learned from experimenting with the R-CC design.

3.1 Data arrival rate measurement. There are two goals for Data arrival rate measurement: good accuracy for reliable congestion detection and low measurement latency for reducing overshooting packets into the queue. To achieve these goals, we started with counting received Data packets in a fixed time interval. However, the discrete packet counting limits the accuracy as packets that arrive around the boundary between two consecutive intervals are unpredictably counted by one of them. We then switched to using a fixed-size sliding window to record data packets' arrival time. Data arrival rate can be estimated by dividing the time difference between the first and last packets' arrival time by the number of packets inside the window. This approach overcomes the accuracy issue in counting packets within a fixed time interval.

3.2 Measure time. Measure time defines the period to measure data arrival rate, as shown in fig 1. Although one must wait for an RTT to start measure the effect of a rate adjustment, our simulation results show that the amount of time needed to get a good measurement accuracy is not tied to RTT but dependent on network traffic dynamics. The independence of measurement period from RTT is important given the lack of a good RTT estimate, it also allows one to balance latency and accuracy by tuning measurement period.

3.3 Using cubic to control interest sending rate. Inspired by TCP CUBIC [2], we increase interest sending rate following a cubic function. When being away from the congestion point, the cubic curve increases fast, enabling quick probing of available bandwidth; when the sending rate gets close to the previous congestion point, the cubic function increases conservatively, reducing queue usage under congestion. We found that the most conservative (flat) region of cubic increases slowly. When congestion happens in the conservative region, consumers may overshoot to the queue for a long time before seeing a noticeable sending-receiving rate mismatch. Thus, we replaced the flat region of cubic with a increasing linear function to specify a minimum interest sending rate increase.

4 PRELIMINARY EVALUATION

We evaluated the design in NDNSim [1], starting from a dumbbell typology with a fixed 1Gbps bottleneck. Five flows with different propagation delays share the same bottleneck link. Fig. 3 shows the interest sending rate of each consumer and the queue usage. The flows converge quickly, and after convergence, the relative difference of the flows' throughput is less than 5%. The bandwidth usage is ~92%, which matches the theory of integrating a cubic curve. Steady-state queue usage is less than 5% of the BDP.

We also evaluated the congestion control under sudden bandwidth changes. The result shows that the consumer can react to congestion caused by bandwidth drops within one round of adjustment (MRTT + measure time). Thus, we believe this design has good potential to work with caching and multipath forwarding.

5 FUTURE WORK

As next step, we will investigate the performance of R-CC in simulation settings with caching and multipath forwarding, and compare R-CC with congestion window-based congestion controls. We also plan to investigate the use of router's congestion feedback at consumers to improve R-CC performance.

ACKNOWLEDGEMENT

This work is partially supported by the National Science Foundation under award 2019012.

REFERENCES

[1] Alexander Afanasyev, Ilya Moiseenko, Lixia Zhang, et al. 2012. ndnSIM: NDN simulator for NS-3. (2012).

[2] Sangtae Ha, Injong Rhee, and Lisong Xu. 2008. CUBIC: a new TCP-friendly high-speed TCP variant. *ACM SIGOPS operating systems review* 42, 5 (2008), 64–74.

[3] Lixia Zhang, Alexander Afanasyev, Jeffrey Burke, Van Jacobson, KC Claffy, Patrick Crowley, Christos Papadopoulos, Lan Wang, and Beichuan Zhang. 2014. Named data networking. *ACM SIGCOMM Computer Communication Review* 44, 3 (2014), 66–73.