PCON as congestion control scheme in NDN

About the paper titled: “A Practical Congestion Control Scheme for Named Data Networking”

Abstract—**Over the past decade, a new network protocol has been on the rise: named data networking, more commonly referred to as NDN. The proposed replacement for the traditional Internet Protocol brings with it a number of benefits, including network-level congestion control. An implementation of such congestion control is PCON, a protocol that monitors outgoing packet queues and prevents congestion through marked packets proposed in the paper ”A Practical Congestion Control Scheme for Named Data Networking”. This essay will provide a brief introduction to named data networking and the structure and rationale of PCON. We will then provide insight into the performance comparison of PCON presented by the authors, upon which we will also critically reflect. We conclude PCON to be a resource-efficient and practical protocol, although we criticize the authors for providing an performance comparison that lacks depth.**

I. Introduction

This essay will focus on a publication by K. Schneider, C. Yi, B. Zhang and L. Zhang [1], in which they introduce PCON, a practical solution for congestion control in named data networking (NDN). The mentioned publication can be divided into two coherent sections, both of which will be discussed in this essay. In the first section, the authors present the reader with an elaborate description of the design specifics of PCON, which they ground in an equally detailed rationale. Here, the focus lies on characteristics of congestion control mechanisms that perform well for Transmission Control Protocol (TCP) with Internet Protocol (IP), but fail to function for NDN. Naturally, insight into the differences between TCP/IP and NDN is required to fully grasp this section. Hence, we will provide a brief introduction to NDN, and highlight once more the differences between it and TCP/IP that are relevant for congestion control.
The second half of the paper puts PCON up against existing solutions for congestion control in NDN.

A. A Brief Introduction to NDN

Named data networking is a network layer protocol, part of a pioneering project within Internet architecture engineering and is one of four projects within NSF’s Future Internet Architecture Program. [5], [6]. A short paper by Afanasyev et al. [2] from the IEEE Xplore Digital Library provides a brief yet effective introduction to NDN, which we shall summarize and link to the course material here.
Whereas traditional “host-centric” internet is centralized around IP and the concept of connections between end devices bearing unique IP-addresses [10], the “data-centric” NDN takes a different approach. The proposed protocol removes the concept of connections (between these end devices) entirely. Instead, NDN uses so-called “named bits”, or “named data”, to serve as the functional cornerstone for locating data. It’s compared to HTTP requests/responses on network granularity level: requests are made by consumers through Interest packets bearing a unique name targeting the data in an application-defined namespace. Such Interest packages are each ultimately replied to with one Data packet containing a meaningful response (or a Negative Acknowledgement (NACK)). The tracking of the data targeted is done through a stateful forwarding plane. Here, NDN nodes have a Pending Interests Table (PIT), a Content Store (CS) and a forwarding strategy with a Forwarding Information Base (FIB), see Figure 1. Upon an incoming Interest packet IN containing the name of some particular data, the node will check its CS, where it may buffer Data packets and which hence could contain the desired data. If the packet cannot be found in the CS, an entry will be made in the PIT containing IN’s incoming and outgoing interfaces. If no entry for IN’s requested data exists already in the table, the node will pass the request to a forwarding strategy, where the FIB is used to determine where (if at all) the desired data can be found. It will then forward the Interest packet to the appropriate interface(s), or respond with a NACK. The direction of this process is referred to as *upstream*. Once an Interest reaches a node containing the desired data, then a data packet D with this data is returned hop-by-hop to all requests over the reverse path of the Interest, where it uses the PIT to decide on the required interfaces. Furthermore, D may be cached in the CS of a node to serve future requests. This process’ direction is called *downstream*.

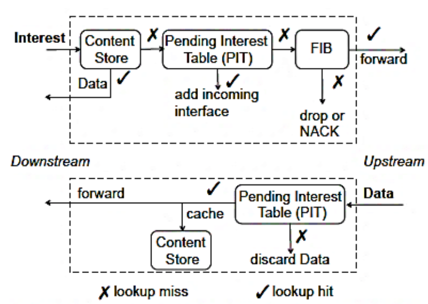

Fig. 1. Forwarding in an NDN node. The *upstream* direction runs from consumer to data repository and *downstream* vice versa

Concerning the network protocol stack, NDN preserves the hourglass shape that we have seen in the course [10]. Here, NDN takes the role of the slim waist, replacing IP.
We will conclude this introduction to NDN by highlighting

the structural differences between it and IP brought about by the above design:

- NDN nodes are not restricted to selecting only one outgoing interface when forwarding. Hence, NDN Interests can have multiple paths and endpoints.
- A consumer can not tell which origin the Data packet has, as an NDN Interest can be answered by any cache that is in the way to the content repository.

The paper by Afanasey et al. also argues explicitly about the benefits and drawbacks of NDN, as well as its support for encryption. However, as only an understanding of the functionality of NDN is needed (and relevant) for this paper, further exploration of these topics is left to the interested reader.

II. The design of PCON

We will discuss four groups of unique NDN characteristics and explain what PCON changed or kept the same so that it works better.

A. Multiple Paths and Endpoints

The most useful characteristic of NDN is, that it has multiple paths and endpoints, which are explained in the introduction. There are some disadvantages to this characteristic that PCON avoids. PCON avoids the risk of a decreased sending rate and unnecessary retransmissions by setting a higher minimum RTO combined with *explicit congestion signaling*. PCON also has a higher avoidance of *cache switchovers*, which could cause an overflow of the queue at the new bottleneck, when the original cache runs out of data and thus cannot answer the next incoming Interests. PCON does this by making buffers large enough to handle a temporary traffic burst at a cache with the CoDel AQM scheme.

B. Pull-based Data Retrieval and HBH Flow Balance

NDN uses Pull-based Data Retrieval, where Data packets follow “breadcrumbs” that are left by Interests. Then NDN uses *Hop-by-Hop Flow Balance* to control the flow with which a given node sends these Interests to another node, thus, in theory, sending exactly one Data packet with each one Interest packet. PCON uses a packet queuing time and local link loss detection as indicators of congestion so that the capacity of a link can be more accurately indicated than an estimation of the link bandwidth or an estimation of the expected load.

C. Diverse Deployment Scenarios

Diverse deployment scenarios are included in NDN. This means that NDN has to work over a range of different deployment scenarios, which include *IP-overlay* and *wireless* links. PCON detects silent drops on both wireless and IP-overlay networks and signals these drops back with explicit NACKs.

D. Flow Semantics and Fairness

NDN uses Flow semantics and Fairness, which makes sure that content is delivered “first come, first serve”. A router does not know whether an Interest comes from a single consumer or an aggregated request of multiple consumers. Hence, if the single consumer comes first, it is given a higher preference than the aggregation that comes later, even if this aggregation might be more important. PCON has no better system in place to solve the level of importance and the level of preference given to different packets. The authors leave this topic to future work.

E. PCON in short

PCON is designed in consideration with the principles of *early congestion detection by adopting an AQM*, *explicit congestion signaling*, *exploiting multipath forwarding* and *special consideration of IP-overlay and wireless links*.
PCON thus consists of five components that improve the NDN architecture.
Firstly it uses Congestion Detection, where each node locally detects congestion by monitoring its outgoing queues [1].
Secondly PCON uses Congestion Signaling, which means that after detecting congestion, nodes will mark Data packets to inform downstream routers and consumers of the congestion [1].
Thirdly it uses Consumer Rate Adjustment, where the end consumers adjust their interest sending rate as a reaction to congestion [1].
Fourthly PCON uses Multipath Forwarding, which makes downstream routers adjust their traffic split ratio as a reaction to congestion signals [1].
Lastly PCON uses Local Link Loss Detection, where it locally detects packet loss on wireless and IP overlay links and signal this packet loss back using NACKs [1].

III. The effectiveness of PCON

PCON is evaluated on different aspects by looking at different implementations and their results in multiple scenario’s. When looking at two aspects, PCON is also compared to other solutions for NDN congestion control from related work.

A. Queue Management

PCON uses CoDel, a scheduling algorithm for controlled delay for it’s active queue management (AQM) [3]. More specifically, PCON uses CoDel dropping queues and explicit congestion signaling [1]. When comparing this method to queue management with first-in-first-out (FIFO) queues and CoDel dropping queues (CODEL_Drop) without explicit congestion signaling the results are that FIFO is problematic due to the fact that it always completely fills the queue and possibly increases the transmission delay. CODEL_Drop is problematic due to the fact that it does not avoid packet drops and thus retransmission delays [1]. PCON does not have these problems.

B. Caching and Time

PCON uses retransmission timeout (RTO) timer settings while CoDel uses TCP timers [3]. Both methods are tested in a scenario where two nodes retrieve the same data. The second node starts later but is able to retrieve data from the cache with higher bandwidth. When the cache is exhausted and both nodes start retrieving the same content, a difference can be noted between PCON and CoDel: CoDel is problematic when the Round-trip Time (RTT) gets small and reduces the RTO to a value smaller than the RTT of the following packets [1]. When this happens, unnecessary retransmissions occur. PCON sets the RTO to a higher value such that $RTO - RTT$ is always positive, preventing unnecessary retransmissions [1].

C. Forwarding Through Multiple Paths

Firstly, PCON is compared to a pending Interest (PI) scheme and a CF scheme [1] [4]. PCON, PI and CF are compared in three different scenarios with three paths; equal bandwidth and RTT, equal bandwidth and different RTT, and different bandwidth and equal RTT. In the first scenario, all reach the optimal split, but in the other two scenarios PI and CF provide a split with less overall throughput due to them preferring the lower-delay path and a bias against high bandwidth paths [1]. Secondly, PCON is compared to the ICP scheme [4]. There are multiple differences between PCON and ICP, the results are summarized below:

- PCON adapts the forwarding ratio based on congestion marks, while ICP does it based on PI. Due to that ICP is faster in adjusting the forwarding percentage [1].
- PCON splits the ratio in a rate that is closer to the maximum bandwidth than ICP [1].
- PCON distributes queuing delay evenly among all bottlenecks while ICP only has queuing delay on the bottleneck of the shortest path [1].

What PCON and ICP do have in common is the fact that their throughput is roughly linear to the RTT ratio [1].

D. High Congestion in Links

In this scenario, signaling a highly congested link (PCON) and 1-Bit congestion marks (CoDel) are compared. It is shown that when a link gets congested 1-bit marking is unable to quickly disable and completely avoid the congested link [1]. This leads to multiple timeouts because it is unknown if there is a congested link [1]. PCON prevents this problem from occurring by disabling it before the timeouts occur [1].

E. IP overlay and Wireless Links

PCON is now compared to another solution; Hop-by-hop (HBH) Interest Shaping. Three different methods are evaluated; The shaper knows the available link capacity in the underlay network, the shaper chooses a parameter value that is higher than the available capacity, and PCON which uses NACKs [1]. The results from the first scenario are that this is an ideal case. In the second case, the queuing delay is high and regular timeouts occur [1]. PCON performs better than the second scenario but is not as good as the ideal scenario [1].

IV. Discussion

We will now discuss the pros and cons of the presented work, along with the things we learned from it and how it relates to the course material.

A. Authors' Goals and Methodology

In II-B it is mentioned that PCON uses a packet queuing time and local link loss detection as an indication of congestion. While measuring the packet queuing time is a good way to reach a higher total throughput, combined with the possibility that it can adapt to the evolution capacity of IP tunnels and wireless links. A downside of this approach is that PCON spends more time when adjusting the transmission percentage to the optimal level to determine the interface that it suits most [8]. In the paper, the authors stress the importance of a fairness scheme. However, such a scheme is never presented and is explicitly left to be developed in future work. Hence, PCON has no notion of packet importance or urgency. In the time past after publication of the paper, an importance scheme for NDN has been published by Zafar et al. [9], which (re)enables PCON to be further developed with regard to fairness and packet importance.

The authors chose to make PCON with the assumption that there are no known link bandwidths or Data packet sizes, as assumptions on these matters may not hold for overlay links, wireless links, or applications with varying packet sizes. This was concluded rightly so, since in practice these parameters may indeed not be known beforehand, especially when applying PCON (or NDN in general) on a large scale on an existing infrastructure.

This idea is reinforced even more by the NSF Future Internet Architecture Project, which backs the NDN project [5]. NSF explicitly deems that its backed projects keep in consideration the social, economic and legal issues they could bring about [5]. If an overhaul of the devices that currently run the Internet would be required, for example, to provide routers with known bandwidth at all times, this would most certainly give rise to such issues and violate this requirement. PCON's ability to run on "any device" makes it a protocol that could be applied to NDN whilst adhering to the aforementioned requirements.

B. PCON as a superior NDN Congestion Control Scheme

In the introduction of this essay, two logical sections of the paper were described. In the second of these sections, as described in III, the authors focus on putting PCON into perspective with its "contenders". As it's only logical for a promising solution to see further development, we observe how PCON has been adopted in research into NDN congestion control schemes.

Doing so, we find that PCON is, in fact, superseded by later development. Indeed, a paper by Thibaud et al. [7] analyzing the challenge of NDN congestion control takes note of PCON as one of a number of possible solutions. In this paper, measurements are done not too dissimilar to those in the work of Schneider et al., and the results are discussed elaborately. It is however in this discussion that other scenarios are examined which give a witness to the fact that PCON does not universally outperform ICP with regard to bandwidth. We shall elaborate on this further:

The paper by Thibaud et al. [7] examines three different scenarios:

- Sce1 considers one consumer and one producer but two paths to examine forwarding through multiple paths.
- Sce2 is a scenario with a bottleneck to examine multipath efficiency and fairness.
- Sce3 is a scenario with a larger topology and can be distinguished in two sub-scenarios:
  - Sce3a has a network with sufficient capacity.
  - Sce3b has a network with insufficient capacity which causes a bottleneck.

Their paper also splits PCON in an end-to-end part for the consumer side (PCON-CS) and a hop-by-hop part for the forwarding strategy (PCON-FS).

To discuss and compare the paper by Schneider et al. and Thibaud et al. [7] we will consider the first of the three scenarios depicted above, as this is most similar to the one presented by Schneider et al., and look at differences and/or similarities between the papers.

In Sec1 The results of the combination ICP and PCON-FS state that "Despite the losses and variable delay, this combination reaches an average rate of 81.3Mbps out of the 100Mbps available [7].", while the combination PCON-CS and PCON-FS has a lower average rate [7]. On the contrary, in III-C it is stated that "PCON splits the ratio in a rate that is closer to the maximum bandwidth than ICP.".

What all three scenarios of the paper by Thibaud et al. [7] have in common is that the original combination, PCON-CS and PCON-FS, limits the use of a second path due to marks [7].

C. Relation to the Course Material

The course material that was presented focused on the implementation of the Internet's transport and network protocol layers that we see today. And, whereas the paper does suggest a congestion control protocol, which has been discussed elaborately in the course, it does so in a completely different setting. In the paper, congestion control runs on the network layer. Particularly, it runs a network layer with NDN rather than IP, which differs vastly and allows congestion control while IP does not, something we have shown earlier in this essay. Logically, this implies that we have learned about this particular new setting, and about this branch of research within Internet architecture engineering.

It is however futile to further compare the means of congestion control for NDN with those that we have learned for IP. That is yet again not to say that the presented work does not relate to the course material at all, rather that the application context of this proposal is fundamentally different from that what was presented in the course. For a comparison of these two contexts grounded in the course material, the reader is referred back to I-A.

V. Conclusion

In conclusion of this essay, we will state what we have learned about the presented work, as well as the context in which it is proposed (data-centric networking). We will reflect on what the authors claim about their proposed solution, and will provide our own opinion about PCON.

A. A "Practical" Solution

The paper presents PCON as a "practical" solution for a congestion control scheme in NDN. We could not agree more on this naming convention. Considering the extended rationale (see in particular II-C, II-D) and explicit absence of assumptions about the implementations PCON is to run on, one can quickly draw the conclusion that this protocol is designed with the aim of running on an arbitrary network device, whilst maintaining reliability. This is something that can be considered obligatory, as NDN, and thus PCON should be able to be implemented without overhauling the entire Internet infrastructure. The latter should hold in accordance with the regulations of the NSF as specified in the discussion.

B. Future Work

As stated in the discussion, a fairness scheme has been made in the years after the paper "A practical congestion control scheme for named data networking" was published. The fairness scheme proposed in [9] can be developed in the PCON scheme. This could bring about great benefits, such as nodes being able to (ab)use the notion of urgency to, for example, efficiently handle the available buffer or table space in the network by prioritizing aggregated Interests (this will free up considerable amounts of space in the PIT's).

We also feel that in further development of PCON, or any congestion control protocol for that matter, the application context should be more elaborately explored. Evidently, as posed in the discussion, Thibaud et al. have been able to present a number of distinct scenarios where PCON does not perform as optimal as the paper by Schneider et al. suggests. We therefore advise that, to be able to develop a (close to) universally efficient congestion control protocol, further work should involve gathering more information about multiple distinct scenarios and different combinations of end-to-end consumer side algorithms and hop-by-hop forwarding strategies, similar to what we see in the work of Thibaud et al.

REFERENCES

[1] Schneider, K., Yi, C., Zhang, B., & Zhang, L. (2016, September). A practical congestion control scheme for named data networking. In *Proceedings of the 3rd ACM Conference on Information-Centric Networking* (pp. 21-30).

[2] Afanasyev, A., Burke, J., Refaei, T., Wang, L., Zhang, B., & Zhang, L. (2018). A Brief Introduction to Named Data Networking. In *MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM)*. doi: 10.1109/MILCOM.2018.8599682

[3] Nichols, K., & Jacobson, V. (2012). Controlling Queue Delay. Queue, 10(5), 20-34. doi: 10.1145/2208917.2209336.

[4] G. Carofiglio, M. Gallo, L. Muscariello, M. Papalini and Sen Wang, "Optimal multipath congestion control and request forwarding in Information-Centric Networks," 2013 21st IEEE International Conference on Network Protocols (ICNP), 2013, pp. 1-10, doi: 10.1109/ICNP.2013.6733576.

[5] NSF Future Internet Architecture Project. (n.d.-b). NSF Future Internet Architecture Project. Retrieved June 9, 2021, from http://www.nets-fia.net

[6] Named Data Networking (NDN) - A Future Internet Architecture. (n.d.). Named Data Networking (NDN). Retrieved June 9, 2021, from https://named-data.net/

[7] Adrien Thibaud, Julien Fasson, Fabrice Arnal, Renaud Sallantin, Emmanuel Dubois, et al. An analysis of NDN Congestion Control challenges. 2nd International Conference on Hot Information-Centric Networking (HotICN 2019), Dec 2019, Chongqing, China. 10.1109/HotICN48464.2019.9063207. hal02314169

[8] Asmaa, E. B., Bouayad, A., & Bekkali, M. E. (2019, October). A hop-by-hop Congestion Control Mechanisms in NDN Networks–A Survey. In 2019 7th Mediterranean Congress of Telecommunications (CMT) (pp. 1-4). IEEE.

[9] Zafar, H., Abbas, Z. H., Abbas, G., Muhammad, F., Tufail, M., & Kim, S. (2020). An Effective Fairness Scheme for Named Data Networking. Electronics, 9(5), 749. MDPI AG. Retrieved from http://dx.doi.org/10.3390/electronics9050749

[10] Ozcelebi, T. & den Hartog, J.I. (2020). Lecture Notes: Computer Networks and Security. Retrieved from 2IC60, Technische Universiteit Eindhoven, https://www.win.tue.nl/∼tozceleb/2IC60/lecture_notes.pdf