2015 IEEE 12th International Conference on Mobile Ad Hoc and Sensor Systems

A Measurement Study on Media Streaming over Wi-Fi in Named Data Networking

Samiuddin Mohammed and Mengjun Xie
Department of Computer Science
University of Arkansas at Little Rock
Email: {stmohammed,mxxie}@ualr.edu

***Abstract*—Named Data Networking (NDN), aka Content-Centric Networking (CCN), excels in content distribution especially multimedia distribution, which can consume significant network bandwidth. With the market penetration of mobile devices and advancement of wireless technologies, media streaming over Wi-Fi becomes increasingly popular but it does not scale well in today's IP based networking. A natural question therefore is how to leverage NDN to improve and optimize media streaming over Wi-Fi. As a first step towards this problem, we set up a 5-node Wi-Fi media streaming testbed based on Wi-Fi Direct technology and use it to collect the bandwidth and CPU usage data when streaming media in NDN as well as in IP networking. We test 4 streaming scenarios in which a live video is streamed from one publisher to multiple consumers over Wi-Fi Direct and present our measurement results in this paper. Our experimental results indicate that the bandwidth consumption between a content publisher and its forwarder (i.e., access point) over Wi-Fi can be effectively and dramatically reduced by NDN, offering much better scalability than IP. However, CPU usage can become much higher in NDN than in IP, which deserves further investigation and optimization.**

***Index Terms*—Named Data Networking, Content-Centric Networking, Video Streaming, Wi-Fi Direct, Performance**

I. Introduction

Recent years have witnessed fast growing demands for media streaming over Wi-Fi, that is, live or prerecorded video and/or audio being transferred from a local source node (either a content publisher or the proxy of a remote source) to one or multiple destination nodes through a wireless access point (AP[1]). For example, users of a home Wi-Fi network who want to watch the same live streaming show from their own device (TV, computer, tablet, etc.) can set up a streaming application (e.g., VLC media player) on a local device to stream the content from its online provider (e.g., YouTube, Netflix) to all the end devices via their home Wi-Fi network, which can effectively save the bandwidth of their broadband connection to the Internet. Video conferencing is another example. Video streaming over Wi-Fi is certainly a better option in terms of quality of service than the content being first forwarded to and then disseminated by a remote server, if all the participants are in the range of the same Wi-Fi network.

Media streaming becomes even more convenient with wide deployment of Wi-Fi Direct compliant mobile devices (e.g., latest smartphone models). Without wireless LAN infrastructure, Wi-Fi devices can still easily form a wireless LAN (WLAN) as long as one of them has Wi-Fi Direct capability. Media streaming over Wi-Fi Direct is particularly suitable for media distribution to a group of users without Wi-Fi infrastructure support. Consider that a group of people want to watch the same football game in a vehicle using their smartphones but there is only one smartphone with a sufficient data plan for the game and the plan is not enough for channel sharing through data tethering. Under this circumstance, the smartphone with the data plan can first form a wireless LAN using Wi-Fi Direct and then stream the game to the other smartphones.

Media streaming over Wi-Fi, however, does not scale well in today's IP networking. Wi-Fi multicast, which can be used to realize IP multicast in a wireless LAN, has several well-known problems [2] and therefore is not suited for media streaming. Although a number of techniques (e.g., [3], [4]) have been proposed to improve multicast over Wi-Fi, they are not widely deployed. A practical solution for media streaming to N nodes over Wi-Fi, which is easy to be adopted by most of Wi-Fi users, is to utilize N unicast sessions instead of setting one multicast session. This approach, however, causes streaming bandwidth to increase linearly with N and leads to poor scalability of the streaming system.

Named Data Network [5] (NDN), or Content-Centric Network (CCN) [6] is a major future Internet architecture that excels in content distribution especially multimedia dissemination. NDN is able to address many issues persisting in IP networking by treating data instead of data's location as the first class entity. Studies have demonstrated that NDN can effectively improve video streaming over the Internet in terms of quality of service and bandwidth saving [7], [8], [9], [10]. A natural question therefore is how to leverage NDN to improve and optimize media streaming over Wi-Fi. Interestingly, there is no systematic study answering this question although it is believed that NDN will help. As a first step towards this problem, we conduct a measurement study to compare IP-based and NDN-based video streaming over Wi-Fi. Our study is different from the work of AMVS-NDN [11] in that we focus on understanding NDN based media streaming over a WLAN in which all the nodes are always within the range of the WLAN and content dissemination only uses Wi-Fi, while AMVS-NDN targets adaptive video streaming in a mobile

[1]The AP can be either a dedicated device in 802.11 infrastructure mode or a "soft" AP (i.e., the group owner of a Wi-Fi Direct network [1]).

978-1-4673-9101-6/15 $31.00 © 2015 IEEE
DOI 10.1109/MASS.2015.56

wireless environment and 3G/4G data offloading and sharing through NDN and Wi-Fi.

We set up a 5-node Wi-Fi media streaming testbed using Wi-Fi Direct technology and employ it to collect the bandwidth and CPU usage data when streaming media in NDN as well as in IP networking. Wi-Fi Direct, initially called Wi-Fi peer to peer (P2P), enables multiple devices to easily form a wireless network without a dedicated wireless AP [1]. We apply Wi-Fi Direct to build the testbed for two reasons. First, Wi-Fi Direct is easy to set up and provides more flexibility than standard 802.11 infrastructure mode. Wi-Fi Direct can be used to emulate a standard WLAN or establish an ad hoc network and the node acting as the soft AP can perform more functions than a standard AP. Second, Wi-Fi Direct is being widely deployed, especially on mobile devices where video is a popular and demanding application.

We test 4 streaming scenarios in which a live video is streamed from one publisher to multiple consumers over Wi-Fi Direct. Streaming scenarios with and without a content forwarder (i.e., AP) are tested in both IP networking and NDN by using VLC media player and NDNVideo [12]. Our experimental results indicate that the bandwidth consumption between a content publisher and its forwarder over Wi-Fi can be effectively and dramatically reduced by NDN, offering much better scalability than IP. We also find that CPU usage can become much higher in NDN than in IP, which deserves further investigation and optimization.

The rest of the paper is organized as follows. Section II briefly presents the background and related work. Section III describes the design of our experiments, followed by Section IV that details experimental results and analysis. Finally, Section V concludes this paper.

II. Background & Related Work

A. Background

Communications in NDN are conducted through two types of packets: Interest packets and Data packets. Both Interest and Data packets carry a name. Forwarding is name-based in NDN rather than address-based in IP. An NDN router employs three functional components for content forwarding: (1) Forwarding Information Base (FIB), (2) Pending Interest Table (PIT), and (3) Content Store (CS). The structure of an NDN FIB is similar to an IP FIB except that NDN allows a match with multiple outgoing links ("faces"). NDN performs a longest prefix match in FIB using name instead of IP address. When an Interest reaches a face of an NDN router, a longest prefix match is performed on its content name. If a Data packet with a prefix match exists in the CS, it will be sent back through the face the Interest arrived on and the Interest will be discarded. Otherwise, the PIT is employed to keep track of pending Interests. The router will first record the content name of the pending Interest and its arriving link in PIT and then forward the Interest upstream toward the origin server(s) hosting the content object. Any router, which has the content object, along the path toward the server will terminate the request and reply with the content object. Thereafter, the content object travels back to the original requestor following the chain of requests recorded in the PITs along the path. When there are multiple pending Interests for the same content, the router only forwards one request (the first one) upstream.

Wi-Fi Direct [1] is a Wi-Fi standard proposed by Wi-Fi Alliance through which a set of devices can form an ad hoc network with a star topology and negotiate which device to be the group owner (GO). The Wi-Fi Direct compliant device designated as the GO essentially acts as a Wi-Fi access point (AP), allowing other devices to connect to it and forwarding traffic. Wi-Fi Direct can operate at higher speeds and greater distances than Bluetooth. It is present in many latest consumer electronics including smartphones, cameras, smart TVs, as well as laptops. An attractive feature of Wi-Fi Direct is that only one Wi-Fi device needs to be compliant with Wi-Fi Direct to establish wireless connections and transfer data. Wi-Fi devices that are not Wi-Fi Direct compliant can still join a Wi-Fi Direct network in the same way it joins a regular WLAN.

NDNVideo [8] is a complete software solution developed for video and audio streaming over NDN. It provides random-access, live, and prerecorded video streaming using a simple data publisher–consumer model. NDNVideo runs NDN over IP. In NDNVideo, a publisher prepares Data packets of video frames and audio samples, signs and stores them in a content repository (or "repo") for retrieval by consumers, whose Interest packets are routed to the repo instead of the publisher. In live streaming a consumer needs to pipeline Interest packets so that Data packets are fetched continuously as they are created. NDNVideo uses a simple packet format for encoded data and transfers audio and video streams separately under different namespaces. A consumer can change video resolution by requesting data from a different namespace for the same time frame. NDNVideo does not implement data deletion or check signatures of successive audio and video packets to confirm the publisher's identity.

B. Related Work

Xu *et al.* compared CCN live streaming to conventional HTTP live streaming on Android and reported that CCN live streaming has a much better scalability [7]. Quereilhac *et al.* proposed a framework for evaluating multimedia applications over information centric network, which is based on the PlanetLab and their developed NEPI experiment management tool [9]. They applied the framework to evaluate CCNx performance in video broadcasting to over 100 nodes. Mangili *et al.* proposed a theoretical framework to compare CCN with CDN (Content-Distribution Network) [13]. They discovered that cache can benefit the Quality of Service (QoS) for both CCN and CDN and that a CDN can provide slightly better performance than a CCN when the total amount of caching storage is the same. From the video streaming experiments over CCN on PlanetLab, Awiphan *et al.* reported that overlay delivery path and data chunk size affect streaming quality [10]. They demonstrated that adaptive bit-rate streaming over CCN could be archived by employing MPEG-DASH. Yuan *et al.*

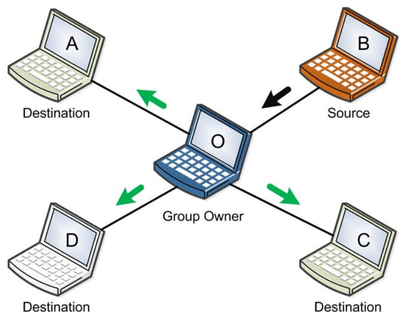


Fig. 1. Scenarios 1 & 2: Video streaming from node B to nodes A, C, and D. The traffic to A, C, and D is forwarded by O. Streaming is based on IP using VLC media player in Scenario 1; Streaming is based on NDN using NDNVideo in Scenario 2.

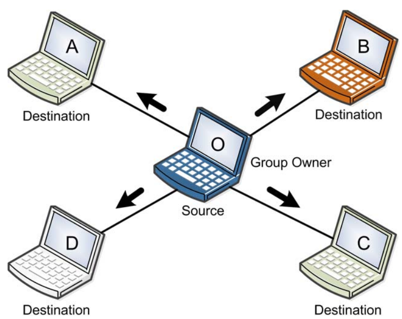


Fig. 2. Scenarios 3 & 4: Video streaming from node O to nodes A, B, C, and D. Streaming is based on IP using VLC media player in Scenario 3; Streaming is based on NDN using NDNVideo in Scenario 4.

compared CCNx and a Web caching based content distribution performance in [14]. Their experimental results indicate that the evaluated CCNx implementation could not sustain gigabit line rates. However, their results also demonstrate the advantages of NDN for networks with lossy access links. With lossy links, CCNx is about 4 times faster than the web caching system. Ciancaglini *et al.* investigated the effectiveness of live TV services in CCN using ccnSim simulator and reported that the PIT is more relevant than caching in reducing server load in live video streaming services [15]. Han *et al.* proposed an adaptive retransmission scheme to recover the video packet losses in content-centric wireless networks in [16].

III. Experiment Design

To emulate a standard wireless LAN (802.11 in infrastructure mode), our testbed is built with a star network topology, in which four nodes are directly connected to the group owner node to form the network based on Wi-Fi Direct. This study focuses on two streaming types: streaming from one source (publisher) to multiple destinations (consumers) with and without a content forwarder (i.e., access point). Four video streaming scenarios, listed as follows, are tested to compare the bandwidth consumption and CPU utilization in an IP network with those in a named data network.

- Scenario 1 (1-B*-IP): IP based streaming from node B
- Scenario 2 (2-B*-NDN): NDN based streaming from node B
- Scenario 3 (3-O*-IP): IP based streaming from node O
- Scenario 4 (4-O*-NDN): NDN based streaming from node O

TABLE I
Devices used in the evaluation

Device(s)	CPU	Memory	Storage	OS
O	Core i5 3210M	4 GB	HDD	Linux 3.10.9 64-bit
B	Core i7 2620M	8 GB	HDD	Linux 3.10.9 64-bit
A & C	Celeron 2955 U	4 GB	SSD	Linux 3.8.2 64-bit
D	Core 2 Duo T7250	2 GB	HDD	Linux 3.10.9 32-bit

Node B as the content publisher streams a video to nodes A, C, and D via node O in scenarios 1 and 2 (Fig. 1), while node O as the content publisher directly streams a video to nodes A, B, C, and D in scenarios 3 and 4 (Fig. 2). Video streaming traffic is carried by IP in scenarios 1 and 3 while it is carried by NDN in scenarios 2 and 4. The content repository is hosted on node O in NDN scenarios. We apply the same testing configurations including topology and hardware to all the scenarios and experiments.

Five laptops are used to build the testbed and their hardware information is listed in Table I. All the testing machines run Ubuntu 12.04. WPA_Supplicant is utilized to set up a Wi-Fi Direct network with those laptops. All the laptops are placed in a research lab and each is within proximity (a few meters) of the others. The impact of location and distance to video streaming performance is left for future study. Nodes A and C are two chromebooks and node D is a fairly old laptop with an Intel Core 2 Duo CPU. We intentionally do not include machines of latest models as we believe the majority of Wi-Fi devices do not have a CPU that is as powerful as the latest models. Given the difficulty of porting the software environment for testing to other types of systems such as Android, we only include laptops as the Wi-Fi node in this study.

We use VLC media player (v2.2) for IP based video streaming in scenarios 1 and 3 and NDNVideo for NDN based video streaming in scenarios 2 and 4. The Real Time Streaming Protocol (RTSP) is used by the VLC media player for video streaming. To run NDNVideo, all the prerequisite packages such as CCNx (v1.0), PyCCN, GStreamer (v0.10) are installed on each node. We set a unique namespace on each node and perform live streaming from the source node's webcam in all the experiments. We apply the same streaming configuration used in [12] (e.g., 704x480 video resolution with H.264 encoding) for both VLC media player and NDNVideo. To collect bandwidth (both inbound and outbound) and CPU load, NetHogs [17] (for measuring bandwidth) and Dstat [18] (for measuring CPU utilization) are also installed on all of the nodes.

For NDN experiments, both CCNx daemon (`ccnd`) and CCNx repository (`ccnr`) processes have to be started before NDNVideo is run. The CCNs repository uses the local file

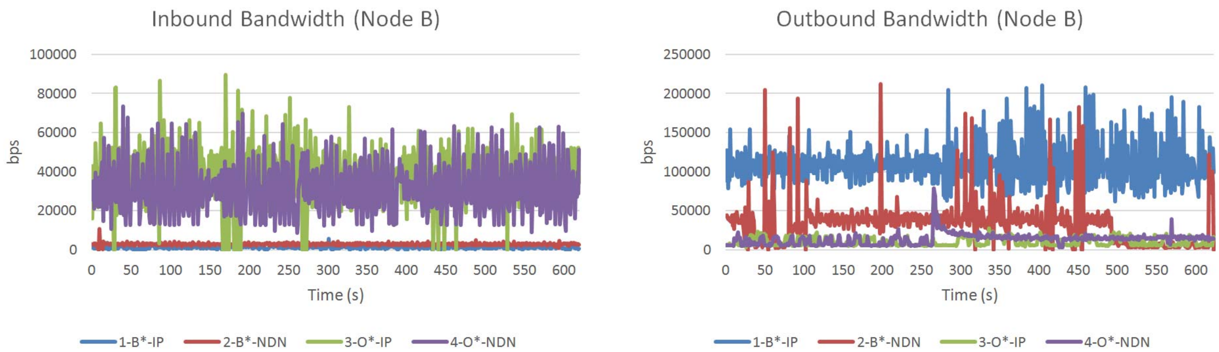


Fig. 3. Inbound (left) and outbound (right) bandwidths of node B

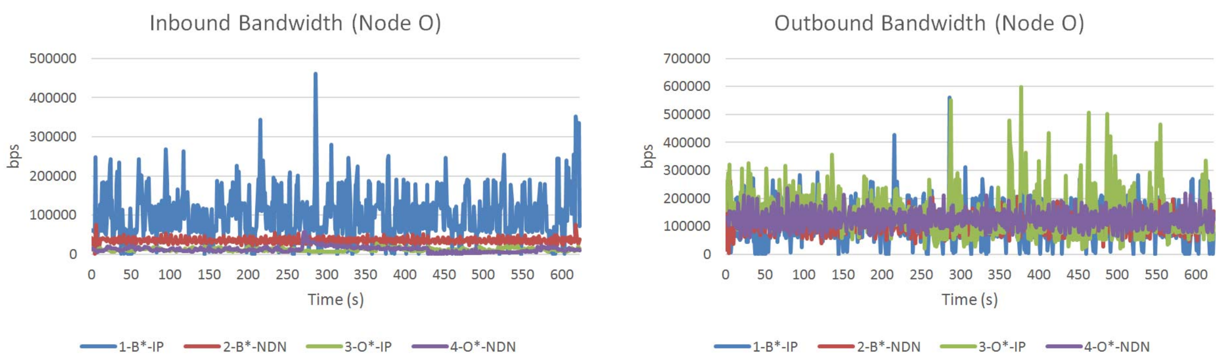


Fig. 4. Inbound (left) and outbound (right) bandwidths of node O

system for persistent storage of CCN content objects and can respond to Interests requesting content that it holds. It is available to CCN components and CCN-enabled applications. The CCNx repository daemon registers itself with the local CCNx daemon so that requests for a namespace are forwarded to the local repository. We run NDNVideo on the publisher node (either B or O) and assign namespaces for audio and video streams. We then open NDNVideo in consumer nodes and request data with the same namespaces as assigned on the publisher node. We use the same FIB entries for destination nodes, which are configured in such a way that it forwards content requests for particular namespaces to node O. All the nodes have fixed Content Store (CS), which is essentially a cache and configured to register the namespace used in evaluation via the content name prefix registration.

IV. Evaluation

We conducted live streaming based experiments for the 4 scenarios a few times using the same environment and configuration. As the results of those rounds are quite similar, we pick the results obtained from one round of experiments that lasted about 620 seconds each and show them in this section. We first present the streaming bandwidth consumption and then the CPU usage of the streaming application measured at each node in the 4 scenarios.

A. Bandwidth Consumption

Node B acts as the content publisher in scenarios 1 and 2 and as a consumer in scenarios 3 and 4. Consequently, the streaming application on B should consume much more bandwidth for outbound traffic than inbound traffic in scenarios 1 and 2 while the inbound bandwidth should be way larger than outbound bandwidth in scenarios 3 and 4. Moreover, for scenario 2, since O hosts the repository that stores the video content from B and all the Interest packets from A, C, and D hit O , duplicate Interest packets will be absorbed by O instead of being forwarded to B. Consequently, only one stream will be established between B and O in NDN while three separate streams will be created from B to A, C, and D in IP. Accordingly, the outbound bandwidth of B in 2-B*-NDN should be significantly smaller than that in 1-B*-IP.

Fig. 3 depicts the inbound (left) and outbound (right) bandwidths consumed by the video streaming on node B in the 4 scenarios. As expected, B consumes very little inbound bandwidth but quite large outbound bandwidth in 1-B*-IP and 2-B*-NDN, and its outbound bandwidth in 1-B*-IP is much higher than that in 2-B*-NDN. The inbound bandwidths in 3-O*-IP and 4-O*-NDN are at about the same level and much higher than their corresponding outbound bandwidth. Another interesting observation is that the outbound bandwidth in 4-

TABLE II
INBOUND AND OUTBOUND BANDWIDTHS (KBPS) MEASURED AT EACH NODE

Scenario	Node O				Node B				Node A				Node D			
	Inbound		Outbound		Inbound		Outbound		Inbound		Outbound		Inbound		Outbound	
	Mean*	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
1-B*-IP	102.7	460.8	109.9	561.0	1.2	5.5	108.9	210.5	37.6	87.6	1.3	10.8	33.9	109.6	1.0	3.1
2-B*-NDN	34.8	74.2	106.9	262.9	2.8	10.5	32.6	211.9	38.8	85.3	1.3	3.5	39.6	84.5	1.6	4.3
3-O*-IP	11.8	28.5	142.6	598.6	37.2	89.5	9.9	27.3	36.9	102.6	1.2	3.7	36.3	149.0	1.2	4.2
4-O*-NDN	12.1	56.8	125.4	237.4	31.0	73.4	13.6	78.6	31.8	73.6	1.3	3.9	32.4	75.5	1.3	3.2

*Note: The mean and max here refer to the average and maximal values of the sampling bandwidths that are reported by NetHogs every second. Let T be the duration of experiment and d_i be the sampling bandwidth at the i-th second. Mean $= \frac{1}{T}\sum_{i=1}^{T} d_i$ and Max $= \max\{d_i, i = 1..T\}$.

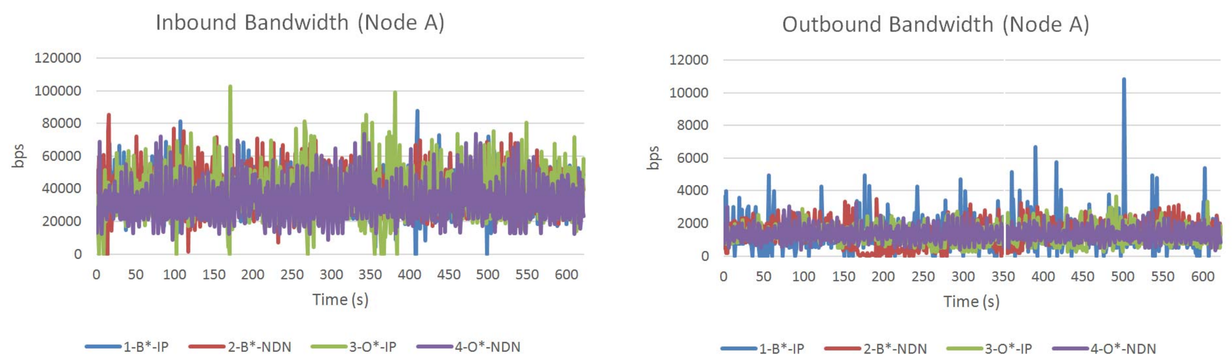


Fig. 5. Inbound (left) and outbound (right) bandwidths of node A

O*-NDN is slightly higher than that in 3-O*-IP, which we believe is mainly attributed to the NDN design that an Interest packet must be issued to retrieve a Data packet. The mean and maximum inbound and outbound bandwidths measured on Node B in each scenario are listed in Table II. The average outbound bandwidths in the 4 scenarios are 108.9, 32.6, 9.9, and 13.6 kbps respectively. The average outbound bandwidth consumed in IP (1-B*-IP) is about 3 times the bandwidth in NDN (2-B*-NDN), demonstrating a significant reduction enabled by NDN.

Node O is the group owner of the Wi-Fi Direct network and plays the role of soft AP. O acts as the content forwarder with 3 streams to A, C, and D in scenarios 1 and 2 and it becomes the publisher with 4 streams to A, B, C, and D in scenarios 3 and 4. Therefore, there should exist significant outbound traffic from O for all the scenarios; In terms of bandwidth, the inbound traffic should be about the same as the outbound traffic in 1-B*-IP (3 streams in, 3 streams out) but only around one third of the outbound traffic in 2-B*-NDN (1 stream in, 3 streams out). The inbound traffic in 3-O*-IP and 4-O*-NDN is expected to be fairly minor.

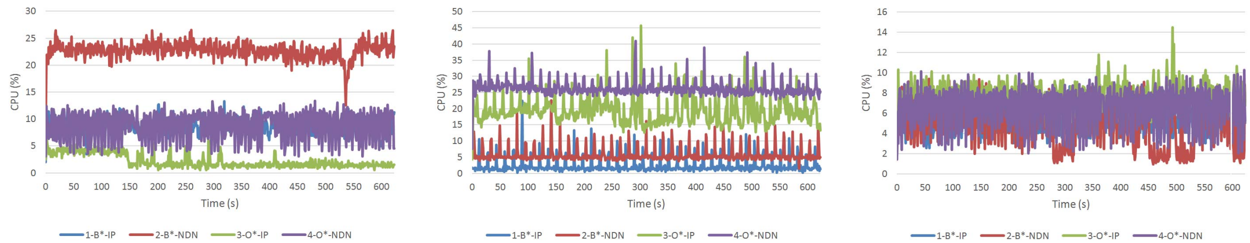
The experimental results shown in Table II have confirmed our estimation. The average inbound and outbound bandwidths in 1-B*-IP are quite close (102.7 kbps and 109.9 kbps) while on average the outbound bandwidth (106.9 kbps) is about 3 times the inbound bandwidth (34.8 kbps) in 2-B*-NDN. Note that for the two parties with direct communication, due to the way the mean bandwidth is derived, the average

TABLE III
CPU USAGE MEASURED (%) ON EACH NODE

Scenario	Node O		Node B		Node A		Node D	
	Mean	Max	Mean	Max	Mean	Max	Mean	Max
1-B*-IP	2.2	22.4	8.6	13.3	5.0	9.1	5.9	12.4
2-B*-NDN	5.7	22.5	22.6	26.5	5.4	9.4	20.3	32.3
3-O*-IP	19.2	45.7	2.2	6.3	6.7	14.5	11.1	18.7
4-O*-NDN	26.2	40.9	9.0	13.4	7.9	10.3	20.1	30.8

inbound bandwidth at one party may not be the same as the average outbound bandwidth at the other party. Node O's average outbound bandwidths in 3-O*-IP and 4-O*-NDN are 142.6 kbps and 125.4 kbps respectively, approximately 4 times the average inbound bandwidths of node B in the two scenarios. The bandwidth dynamics measured on node O in the 4 scenarios are displayed in Fig. 4.

Nodes A and C have the same hardware and software configuration and play the same role in the experiments. Thus, their results are pretty much the same and we only report the mean and maximal bandwidths of A in Table II and display bandwidth dynamics in Fig. 5. A acts as a consumer in all the 4 scenarios. Therefore, A's outbound traffic is essentially negligible compared to its inbound traffic, as shown in the table. The average inbound bandwidths measured on A in the 4 scenarios are at the same level, also as expected. Due to space limitation, we only include the mean and maximal bandwidths of D in Table II but do not show the figure of D's bandwidths, which is very similar to Fig. 5.



(I) Node B (II) Node O (III) Node A

Fig. 6. CPU usage measured on each node

B. CPU Utilization

We collected CPU usage (per second) data of the streaming application on each node using dstat. Fig. 6 depicts the CPU usage on nodes B, O, and A and Table III lists the mean and maximal CPU usage data on B, O, A, and D for the 4 testing scenarios. Due to the hardware difference, the CPU usage data of two different nodes should not be directly compared for each scenario. From the CPU usage data listed in Table III, it is evident on all of those nodes that the streaming application on average consumes more CPU time in an NDN scenario than in its corresponding IP scenario. The higher CPU usage in NDN is attributed to the cost in managing NDN specific components including the content repository, CS, PIT, and FIB, which occurs on every node no matter that node is a publisher or a consumer. An interesting observation is that the gap of CPU load between NDN and IP is much smaller on node A (and C), which may be attributed to the special configuration on those two Chromebooks which we had to make for running Ubuntu and NDNVideo.

For node B, the highest CPU usage occurs when B is the publisher in NDN, which in understandable since B has to maintain streaming service and NDN repo and tables at the same time. For node O, its CPU load is fairly small when O plays the role of content forwarder. Although O has to forward all the video and audio data to A, C, and D, much of the computation in forwarding can be offloaded to its network card. As the CPU on node D is quite old, more CPU time has to be spent on the video streaming application and NDN maintenance, rendering higher CPU usage than other consumer nodes.

V. Conclusion

Video streaming over Wi-Fi is getting more popular but faces a big challenge in high network bandwidth usage. NDN has a great potential to address this issue given its innate advantage in content distribution. In this paper we presented a measurement study on video streaming over Wi-Fi Direct, in which we compared the bandwidth and CPU usage of a live streaming session using today's IP networking and proposed named data networking. Our experiment results have confirmed the superiority of NDN in bandwidth saving for content distribution over Wi-Fi. Our study also reveals that the CPU usage of video streaming can become much higher in NDN than in IP. The results suggest further optimization in the implementation of NDN content repository and major components (CS, FIB and PIT) as well.

References

[1] Wi-Fi Alliance, "Wi-Fi Direct," http://www.wi-fi.org/discover-wi-fi/wi-fi-direct.

[2] D. Dujovne and T. Turletti, "Multicast in 802.11 wlans: An experimental study," in *Proceedings of the 9th ACM International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '06, 2006, pp. 130–138.

[3] R. Chandra, S. Karanth, T. Moscibroda, V. Navda, J. Padhye, R. Ramjee, and L. Ravindranath, "Dircast: a practical and efficient wi-fi multicast system," in *Proc. ICNP*, 2009, pp. 161–170.

[4] S. Sen, N. K. Madabhushi, and S. Banerjee, "Scalable wifi media delivery through adaptive broadcasts." in *NSDI*, 2010, pp. 191–204.

[5] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang *et al.*, "Named data networking," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 66–73, 2014.

[6] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proc. CoNEXT*, 2009, pp. 1–12.

[7] H. Xu, Z. Chen, R. Chen, and J. Cao, "Live streaming with content centric networking," in *Networking and Distributed Computing (ICNDC), 2012 Third International Conference on*. IEEE, 2012, pp. 1–5.

[8] D. Kulinski, J. Burke, and L. Zhang, "Video streaming over named data networking," *IEEE COMSOC MMTC E-letter*, vol. 8, no. 4, 2013.

[9] A. Quereilhac, D. Camara, T. Turletti, and W. Dabbous, "Experimentation with large scale icn multimedia services on the internet made easy," *IEEE COMSOC MMTC E-Letter*, vol. 8, no. 4, pp. 10–12, 2013.

[10] S. Awiphan, T. Muto, Y. Wang, Z. Su, and J. Katto, "Video streaming over content centric networking: experimental studies on planetlab," in *Proc. Computing, Communications and IT Applications Conference (ComComAp)*, 2013, pp. 19–24.

[11] B. Han, X. Wang, N. Choi, T. Kwon, and Y. Choi, "AMVS-NDN: Adaptive mobile video streaming and sharing in wireless named data networking," in *Proc. INFOCOM Workshops*, 2013, pp. 375–380.

[12] D. Kulinski and J. Burke, "Ndnvideo: Random-access live and pre-recorded streaming using ndn," UCLA, Technical Report NDN-007, Sept. 2012.

[13] M. Mangili, F. Martignon, and A. Capone, "A comparative study of content-centric and content-distribution networks: Performance and bounds," in *Proc. GLOBECOM*, 2013, pp. 1403–1409.

[14] H. Yuan and P. Crowley, "Experimental evaluation of content distribution with ndn and http," in *Proc. INFOCOM*, 2013, pp. 240–244.

[15] V. Ciancaglini, G. Piro, R. Loti, L. A. Grieco, and L. Liquori, "Ccn-tv: a data-centric approach to real-time video services," in *Proc. Advanced Information Networking and Applications Workshops (WAINA)*, 2013, pp. 982–989.

[16] L. Han, S.-S. Kang, H. Kim, and H. P. In, "Adaptive retransmission scheme for video streaming over content-centric wireless networks," *IEEE Communications Letters*, vol. 17, no. 6, pp. 1292–1295, 2013.

[17] A. Engelen, "NetHogs," http://nethogs.sourceforge.net/.

[18] D. Wieers, "Dstat: Versatile resource statistics tool," http://dag.wiee.rs/home-made/dstat/.