Enhancing Blockage Detection and Handover on 60 GHz Networks with P4 Programmable Data Planes

Ali Mazloum*, Elie Kfoury*, Sanjib Sur*, Jorge Crichigno*, Nasir Ghani†
*College of Engineering and Computing, University of South Carolina, Columbia, U.S.A.
†Department of Electrical Engineering, University of South Florida, Tampa, U.S.A.
*amazloum@email.sc.edu, *ekfoury@email.sc.edu, *sur@cse.sc.edu, *jcrichigno@cec.sc.ed, †nghani@usf.edu

***Abstract*—Millimeter-Wave (mmWave) technology for 5G networks is promising due to its capability of supporting high data rates and low-latency communications. Despite such benefits, its narrow beams make it highly susceptible to blockages. To overcome blockages, solutions rely on handover connections from the current (blocked) access point to an alternative (non-blocked) access point. With state-of-the-art techniques, handover decisions may take hundreds of milliseconds to a few seconds at best, thus disrupting connections, dropping packets, and substantially reducing throughput.**

This paper proposes a P4 Programmable Data Plane (PDP) assisted system that reduces the handover time between access points when a blockage occurs. The proposed scheme relies on the capability of a PDP to track flows in real-time and measure network metrics with nanosecond resolution. The PDP computes the packet Inter-Arrival Time (IAT) for each flow that is routed through a mmWave access point. As a blockage occurs, the scheme is able to detect sudden increases in the packet IAT and handover flows to a non-blocked access point. Evaluation results show that the proposed approach can detect blockages faster than current solutions, thus mitigating the corresponding throughput degradation and packet losses. On average, the proposed approach can detect blockages in 20 milliseconds.

***Index Terms*—P4, Programmable Data Planes (PDP), mmWave, 5G, fast rerouting.**

I. INTRODUCTION

To support the multi-fold increase in demand for mobile data, the upcoming wireless systems are expected to provide multi gigabits per second (Gbps) rates, well above the existing Wi-Fi technologies. The mmWave technology is emerging as the most promising solution to meet the traffic surge. It operates on less-congested spectrum bands of Extremely High Frequencies (EHFs), between 30-300 gigahertz (GHz), enabling multi Gbps throughput [1].

Despite its potential, mmWave signals suffer from high sensitivity to blockage due to the use of high frequency and narrow (highly directional) beams. Communication over mmWave is characterized by a quasi-optical propagation behavior [2] where the received signal is dominated by the line-of-sight (LOS) path and first-order reflections from strong reflecting materials. However, the LOS path might be easily blocked, even by human bodies. Upon blockage, the throughput of a mmWave connection might be highly affected due to the degradation in the physical channel quality. For instance, blockage by a human can penalize the link budget by 20-30 dB [3], resulting in a sudden outage of the received signal [4].

Densified access point deployment and connection handover upon blockage detection are used as a potential solution [5, 6]. By performing handover, the LOS path of multiple access points can be leveraged to sustain a high-performance connection. The effectiveness of this approach increases as the number of deployed access points increases. However, detecting the time for the handover to be triggered (known as the handover decision problem) is not straightforward.

Multiple researchers proposed proactive approaches that utilize camera images or the end user's location to predict and react to a blockage before it occurs [7, 8]. Other researchers proposed offloading the handover decision to a centralized controller, which monitors the throughput and initiates the handover when the throughput degrades [9]. However, the contribution of the reactive approaches that monitor the traffic (e.g., throughput) to perform handover is limited, as they require at least one second to detect the blockage [9]. Such slow reaction results from the low performance of controllers as the traffic load increases.

With the emergence of data plane programmability, network forwarding devices become capable of providing nanosecond granularity, per-packet visibility, and performing line-rate actions, while sustaining terabits per second processing speed. As a result, an increased number of applications, from rerouting for load balancing [10] (e.g., utilization minimization [11, 12]) to TCP optimization [13] have been proposed.

This paper proposes offloading the handover decision to a Programmable Data Plane (PDP) switch in a mmWave network. By utilizing the per-packet visibility and the line rate actions provided by PDP switches, handover decisions can be made in a few milliseconds (ms) after blockage occurrence. Intuitively, when a blockage occurs, the packets will not be delivered to the receiver due to a disruption in the physical layer, which causes the packet's Inter-Arrival Time (IAT) to increase significantly. PDP switches can infer the blockage by detecting such an increase in the packet's IAT. Evaluation results show that during the blockage, the IAT of a packet is in order of hundreds of milliseconds and that the proposed system can detect blockage in 20ms, on average. The contributions of

---

This work was supported by the U.S. National Science Foundation, Office of Advanced Cyberinfrastructure, Award #2118311.

this paper are:

- Leveraging PDP switches to compute the packet's IAT and detect mmWave blockage on a per-packet basis.
- Conducting evaluations on a testbed composed of real devices, including a widely used PDP switch (Intel's Tofino), mmWave access points, and a mobile device operating in the 60 GHz band.
- Detecting the blockage and initiating handover before the throughput degrades from the blockage.
- Proposing a solution to the handover decision problem without modifying end devices and extending the system to select the best alternative non-blocked access point by utilizing information from end devices.

II. RELATED WORK

Mezzavilla et al. [14] proposed an optimal cell selection in mmWave networks. They used dynamic programming to detect blockage proactively. Dhahri et al. [15] and Tabrizi et al. [16] considered both the current state and possible future states of the network for handover decisions. They presented a reinforcement learning framework that explores the past cell behaviors of a target cell and predicts its future state based on the Q-learning algorithm [17].

Zang et al. [7] leveraged the user mobility information and predicted the future data rates when a user enters an area blocked by a static obstacle. Koda et al. [18] extended this approach and considered the future degradation that a moving obstacle might cause. Oguma et al. [8] proposed a camera-based framework in mmWave networks. Images from the camera were used to detect the occurrence of the blockage proactively. The camera detected if a human was approaching the LOS path between the access point and the user device and rerouted flows accordingly.

Singh et al. [3] and Song et al. [19] designed reactive communication control schemes that detect the blockage based on the quality of the channel. The channel quality is assessed by many factors, including the received signal strength indicator (RSSI) and frame loss rate. Tsang et al. [20] proposed a reactive approach that leverages the non-LOS (NLOS) paths through beamforming to alleviate the blockage problem.

A common drawback of these approaches is the need for continuous input from external sources such as a camera, an access point, or user equipment (UE). Relying on information from external sources requires fundamental changes to the current mmWave infrastructure and limits the range of deployment. Moreover, most of these approaches either simulate the testing environment or use a modified access point.

Oguma et al. [9] proposed using an external controller that monitors the throughput of a mmWave connection. When the throughput drops below a specific threshold, the system assumes that a blockage occurred and reroutes the traffic. Overall, the system requires at least one second to recover from a blockage, which is not acceptable for many real-time applications.

In contrast, our proposed approach can detect blockages within 20ms. It relies on PDP switches to measure network metrics according to a customized P4 program. The switches calculate a packet IAT in the data plane in real-time and use it to detect blockages and handover connections (no external inputs are required to detect the blockage). Besides, the proposed approach can be extended to select the best alternative access point by leveraging RSSI measurements from a UE.

III. BACKGROUND

A. Handover Decision Problem

A mmWave band, such as 60 GHz, can suffer from 20 to 30 dB more attenuation than traditional Wi-Fi bands at 2.4 or 5 GHz. This loss can be compensated by focusing the transmission and reception energy within narrow beams using multiple phased-array antennas. The sender and the receiver align their narrow beams to maximize the signal-to-noise (SNR) ratio and the channel capacity. Under this technique, the received signal becomes dominated by the LOS path and first-order reflections from strong reflecting materials. Thus, using the LOS path is essential to achieve a high-performance connection. However, the LOS path might be easily blocked, even by human bodies, resulting in an unreliable connection with a significantly degraded throughput.

Fortunately, the LOS paths from multiple access points can be leveraged to mitigate the blockage. This requires a densified access point deployment, such that when a blockage occurs, the user equipment (UE) handovers to an alternative access point with a corresponding clear LOS path [5, 6]. An open research issue is to detect when such handover should be performed.

B. PDP and P4 Language

A PDP enables the programmer to develop customized code that takes actions on network packets at line rate. It also provides granular visibility of events occurring in the data plane, reduces complexity and enhances resource utilization, and drastically improves the performance of applications that are offloaded to the data plane [21]. The de-facto language for describing the data plane behavior is Programming Protocol-independent Packet Processors (P4), which is a domain-specific language for networking. The P4 compiler accepts as input a P4 source code and generates a binary code that is loaded into the data plane.

The key features used by our proposed scheme are the per-packet visibility and the offloading of detection functionality to the data plane. To obtain visibility, we measure network statistics with high precision using a PDP switch. Moreover, to detect blockages, we track flows and use network statistics to identify pattern deviations using a PDP switch. Therefore, computation in the data plane does not incur a performance penalty.

IV. PROPOSED SYSTEM

A. Overview

The proposed scheme offloads the handover decision upon blockage from the control plane to the data plane. The

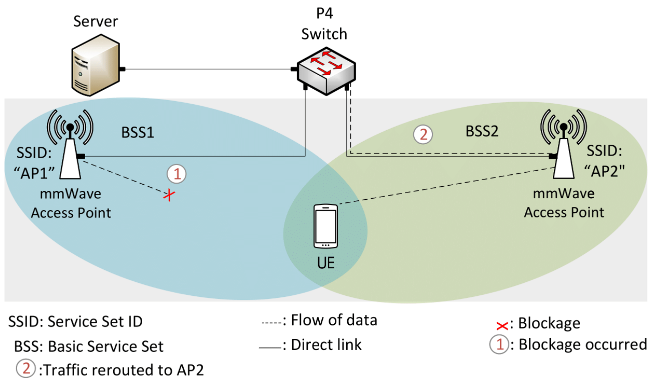


Fig. 1. System overview. Data traffic is generated between the server and the UE. Step 1: the LOS path between the access point currently used to connect to the network and the UE is blocked. Step 2: the PDP switch detects the blockage, reroutes the traffic to the alternative access point, and notifies the UE to handover.

blockage event can be characterized by tracking flows and computing their corresponding packets IAT. Packet IAT is the time duration that separates the arrival of two consecutive packets of the same flow.

Consider Fig. 1. The UE uses the air interface to connect to an access point. The carrier frequency is in the mmWave band. Access points are connected to the network infrastructure through the PDP switch. Any degradation in the quality of the physical channel is reflected as an increase in the IAT values. The experiments show that when the LOS path between the access point and the UE is blocked, the IAT of consecutive packets increases. This increment is detected by the switch. Thus, by computing the packet IAT, the PDP switch can detect the occurrence of a blockage.

B. *Assumptions and Requirements*

1) Using the 2.4 GHz channel to exchange control messages between the UE and the PDP switch: An access point can maintain two independent wireless connections with a UE, one connection operating at 2.4 GHz (or 5 GHz) and the other operating at 60 GHz. Communication over 2.4 GHz and 5 GHz are significantly more resilient to blockage compared to communication over 60 GHz. In the proposed system, the 2.4 GHz channel is utilized by the UE and the PDP switch to exchange control messages. The UE uses the channel to periodically update the PDP switch with the SSID of the alternative access point having the highest RSSI. The PDP switch uses the 2.4 GHz channel to notify the phone to perform handover when the 60 GHz link is blocked. The notification contains the SSID of the alternative access point.

2) Identifying IAT increases caused by non-blockage events: For the preliminary results reported in this paper, the sender (server) is continuously sending data to the receiver (UE). Under that condition, it suffices to monitor the IAT of packets coming from the server, as a significant increase in the IAT will be only caused by LOS blockage. Future work aims at considering different data patterns by monitoring the time difference between receiving a packet from the server and receiving the corresponding acknowledgment from the UE. Any increase in the time to acknowledge packets is guaranteed to be a consequence of the LOS blockage.

Algorithm 1: IAT estimation and blockage detection

```
Control plane():
    if $digest$ then
        $flow\_ID \leftarrow digest.flow\_ID$;
        $pkt = new(Packet)$;
        $pkt.probe.flow\_ID \leftarrow flow\_ID$;
        $send\_packet\_every\_T\_ms(pkt)$;
Data plane():
    $hdr \leftarrow pkt.extract(headers)$;
    if $hdr.probe.inValid()$ then
        $flow\_ID \leftarrow hash(5\_tuple)$;
        if $register[flow\_ID] == 0$ then
            $send(digest, flow\_ID)$;
        $register[flow\_ID] \leftarrow time.now()$;
    else
        $flow\_ID \leftarrow hdr.probe.flow\_ID$;
        $last\_pkt\_AT \leftarrow register[flow\_ID]$;
        $IATe \leftarrow time.now() - last\_pkt\_AT$;
        /* detection_threshold is hardcoded */
        if $IATe > detection\_threshold$ then
            Update the output port;
            Notify the UE;
```

C. *IAT Estimation and Blockage Detection*

Experiments showed that the 60 GHz link might require hundreds of milliseconds to deliver a packet during the blockage. Thus, even if the PDP switch detects the blockage from the IAT of the first packet traversing the blocked 60 GHz link, the detection speed will be in the order of hundreds of milliseconds. To enhance the detection speed, the PDP switch is programmed to estimate a lower bound on the IAT of the packets. The IAT values are estimated on a per-flow basis. Each data flow is uniquely identified by the hash of the 5-tuple: source IP address, source port, destination IP address, destination port, and transport layer protocol.

Fig. 2 summarizes the IAT estimation process. The server initiates a data flow that generates data packets. The data plane notifies the control plane that a new flow is detected. The control plane then initiates a monitoring flow. The probes generated by the monitoring flow are separated by a time gap t. For each incoming probe, the data plane computes the time difference between the arrival time of the probe and the arrival time of the last packet observed from the data flow. The time difference is a lower bound estimation for the IAT value of the data packet that will arrive after the probe. The PDP switch uses the estimated IAT (IATe) to detect blockage.

The pseudocode of the PDP program is summarized in Algorithm 1. The data plane notifies the control plane about a new flow by sending a digest containing the ID of the flow. The control plane extracts the ID from the notification and initiates a monitoring flow. The packets of the monitoring flow include the ID of the flow being monitored inside a custom header.

On the data plane, the parser extracts the header fields of an incoming packet. A packet that does not include the custom header (hdr.probe) is processed as a data packet. The hash of

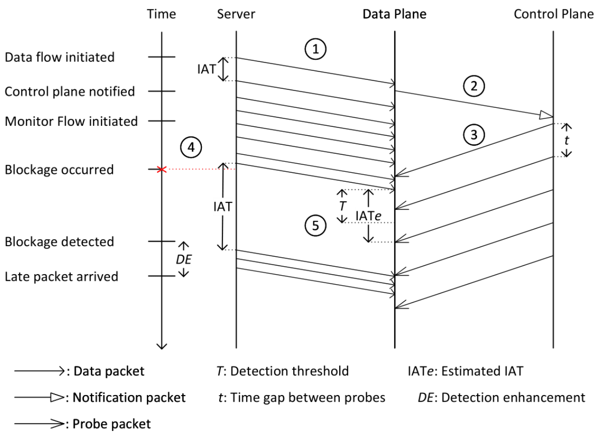


Fig. 2. IAT estimation scheme. Step 1: the server initiates a data flow. Step 2: the data plane notifies the control plane that a new flow is detected. Step 3: the control plane initiates a monitoring flow. Step 4: blockage occurred. Step 5: the blockage is detected as the estimated IAT value (IATe) is larger than the detection threshold (T).

a data packet is calculated using its 5-tuple. If the hash value points to an empty register entry (stateful memory storage in the data plane), the data plane notifies the control plane that a new flow is detected. Otherwise, the value at the register entry is updated to be the arrival time of the current packet.

However, if the incoming packet contains the custom header, the flow_ID is extracted. The last arrival time of the flow to which the extracted flow_ID maps is retrieved from the register array. The switch calculates the time difference between the arrival time of the current probe and the retrieved arrival time. This time difference represents the estimated IAT ($IATe$). The $IATe$ value is then compared against a blockage detection threshold (detection_threshold). If there is a blockage, the $IATe$ should be larger than the threshold. In such a scenario, the switch reroutes the traffic to the alternative access point and uses the probe to notify the UE to handover.

V. EXPERIMENTATION

A. Topology Setup

Consider Fig. 3. The testing topology consists of a server, a PDP switch, two mmWave access points, and a UE. The PDP switch connects the access points to the server and computes the IAT values. The access points are Netgear NightHawk x10 [22]. The UE is ASUS ROG phone [23]. The phone and the mmWave access points use IEEE 802.11ad standard to communicate. The distance between the phone and each access point is 2 meters, as suggested by previous work [24]. The phone is connected to the two access points simultaneously using reverse tethering. In reverse tethering, the access point is configured to be in the managed mode, and the phone is configured to be in the access point mode. This technique eliminates the overhead of re-authentication and re-association when the phone handover to the non-blocked access point, thus achieving access point switching in units of milliseconds [25]. The concept of connecting one device to multiple access points is not new, as it is supported by many legacy 802.11 drivers. The latest driver for 60 GHz wireless adapters might adopt this functionality in future versions [26]. iPerf3 tool [27] is used by the server to generate traffic.

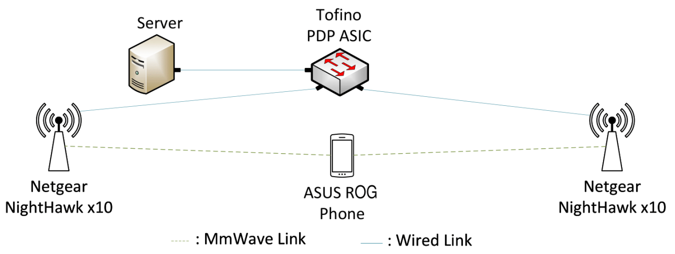


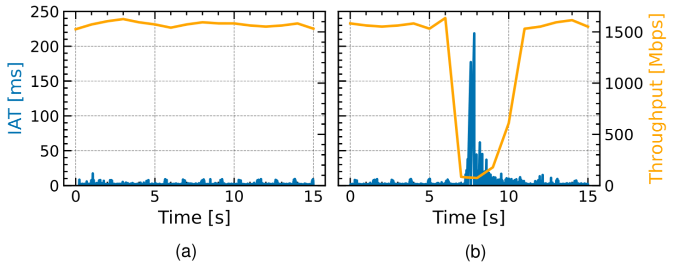
Fig. 3. Testbed used for the experiments.

B. Traffic Analysis

To identify the effect of blockage on the IAT of packets, the IAT values during a blockage are compared to the IAT values during a clear LOS transmission. The experiments are repeated 60 times and the results are averaged. In clear LOS, the average IAT is 7 microseconds (us). The standard deviation is 64us, which indicates a high fluctuation in the IAT of the packets. Around 99.97% of the packets' IAT values are less than 1ms, out of which 93.3% are less than 1us. The maximum observed IAT value is 18.8ms. This multi-fold increase in the IAT value is either due to a change in the modulation and coding scheme (MCS) [28] or due to a change in the signal amplitude [29]. The MCS change is caused by the variation in the signal strength, whereas the signal amplitude change is caused by beam realignment [30].

During blockage, the average IAT of packets has significantly increased. The maximum observed IAT values are 24,200x and 32,800x larger than the average IAT, respectively. A comparison between the IAT in a clear LOS and during a blocked LOS is depicted in Fig. 4 at which (a) represents an experiment with clear LOS and (b) represents an experiment where the LOS is blocked at t=7s for 2 seconds.

The proposed system detects blockage by comparing the IAT values with a fixed threshold. The threshold should minimize both the blockage detection speed and the number of false detection. To this end, the threshold should be slightly larger than the maximum observed IAT in a clear LOS



(a)

(b)

Fig. 4. Blockage can be identified by the significant increase in the IAT values during the blockage: (a) no blockage; (b) blockage at t=7s.

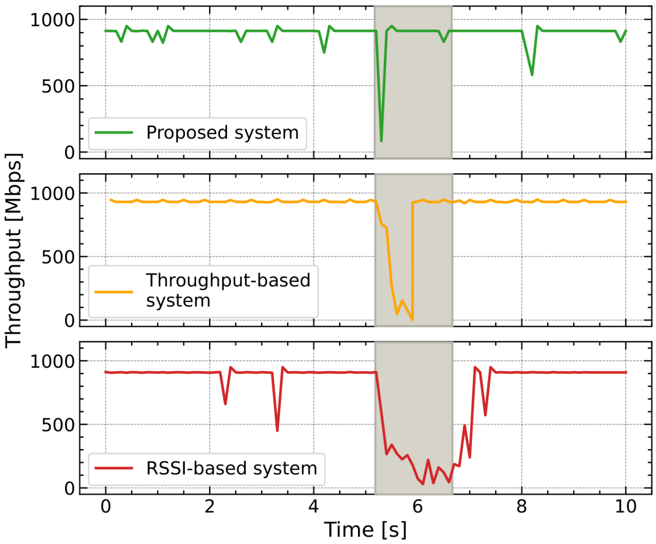


Fig. 5. Recovery speed of the proposed system, throughput-based system, and the RSSI-based system. The gray rectangle represents the 2 seconds blockage.

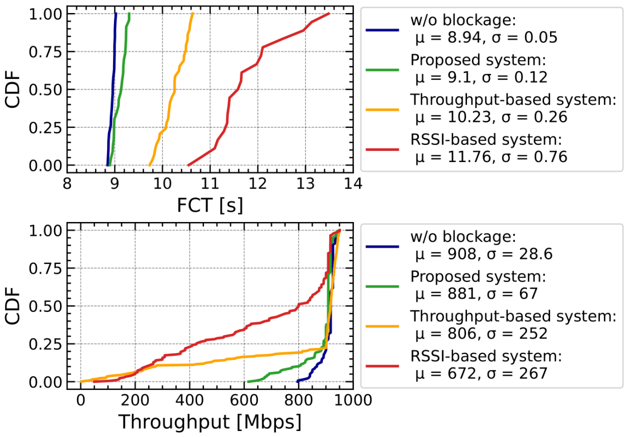


Fig. 6. Cumulative distribution function of the flow completion time (FCT), and the throughput of the w/o blockage scenario, blockage mitigated with PDP system scenario, blockage mitigated with throughput-based system scenario, and blockage mitigated with RSSI-based system scenario.

transmission. The maximum observed IAT value was 18.8ms, so the threshold was chosen to be 20ms. To evaluate the threshold, 60 experiments were performed in a clear LOS, and the 20ms threshold was applied. In this set of experiments, the number of false detections was zero. Another 60 experiments were performed where the LOS was blocked. The switch successfully reported the blockage in all the experiments.

C. Proposed System Evaluation

1) Throughput degradation: Following a similar evaluation approach to [24], the maintained throughput during blockage is used to evaluate the proposed system. The performance of the system is compared to two reactive systems: (1) a conventional system that monitors the RSSI to perform handover; and (2) a system that monitors the throughput to perform handover. The conventional system deployed by ASUS ROG phone automatically initiates the handover when the RSSI drops below -75 decibels relative to one milliwatt (dbm) for 5 seconds (the values are determined by inspecting the source code of the ASUS ROG phone's kernel which can be found at [31]). The throughput-based system proposed by Oguma et al. [9] uses a 40% degradation in throughput as a threshold to detect blockage, where the controller initiates the handover when the monitored throughput degrades to be less than the threshold. The system proposed by our work uses a 20ms detection threshold and a 2ms probe generation gap.

Three experiments are conducted where the PDP-based handover mechanism is deployed in the first experiment, the RSSI-based handover mechanism is deployed in the second experiment, and a controller that monitors the throughput and initiates the handover is implemented in the third experiment. The controller is connected with the first mmWave access point using a 1Gbps Ethernet cable. Because the throughput in the third scenario is constrained by the 1Gbps cable, the throughput of the three experiments was limited to 950Mbps so that the performance of the three experiments could be compared. In the three experiments, the LOS is blocked at t=5s for 2 seconds. The blockage duration is 20% of the duration of the test, which is adopted from [24].

Consider Fig. 5. The gray rectangle indicates when the blockage occurred and the duration of the blockage (the switch is used to report the start time of the blockage). As shown by the figure, the system detects the blockage before the throughput degrades. The proposed system outperforms the throughput-based system and the RSSI-based system, at which the recovery speed of the proposed system is 5x faster than the throughput-based system and 10x faster than the RSSI-based system. The proposed system requires around 200ms to fully recover from the blockage; the throughput-based system requires around 1s to recover from the blockage, and the RSSI-based system requires around 2s to recover from the blockage.

2) Flow completion time: This set of experiments evaluates the flow completion time (FCT) of a 1 gigabyte flow in four different scenarios. In the first scenario, the LOS is not blocked. This scenario represents the solution at which blockage detection and traffic handover are totally seamless. In the other scenarios, the LOS path is blocked for 2 seconds, such that the proposed system is deployed in the second scenario, the throughput-based system is deployed in the third scenario, and the RSSI-based system is deployed in the fourth scenario. Each scenario was repeated 50 times.

Fig. 6 shows the Cumulative Distribution Function (CDF) of the FCT in the four scenarios, and the corresponding CDFs of the throughput. For the w/o blockage scenario, the average FCT is 8.94s, and the standard deviation is 0.05s. In the scenario deploying the throughput-based system, the average FCT is 10.23s, which is 1.29s slower than the w/o blockage scenario. The standard deviation is 0.26s, which is 5.3x larger than the standard deviation of the w/o blockage scenario. In the scenario deploying the RSSI-based system, the average

FCT is 11.76s, which is 2.82s slower than the w/o blockage scenario. The standard deviation is 0.76s, which is 15.2x larger than the standard deviation of the w/o blockage scenario. In the proposed system, the average FCT is 9.1s, which is 0.16s slower than the w/o blockage scenario, 1.13s faster than the throughput-based system, and 2.26s faster than the RSSI-based system. The standard deviation is 0.12s, which is 2.4x larger than the standard deviation of the w/o blockage scenario, 2.16x smaller than the standard deviation of the throughput-based system, and 6.3x smaller than the standard deviation of the RSSI-based system.

VI. CONCLUSION

This paper proposes a novel way to detect blockage events. The scheme leverages the per-packet granularity of PDP switches to accurately detect the disruptions in the physical layer caused by a blockage occurrence. All the testing experiments are conducted on real data using off-the-shelf mmWave compatible devices. Testing results show that the proposed system needs a threshold in order of milliseconds to detect the blockage. The system can utilize RSSI measurements from end devices to solve the access point selection problem. Future work aims at considering different traffic patterns by producing a dynamic blockage threshold.

REFERENCES

[1] P. Wang, Y. Li, L. Song, and B. Vucetic, "Multi-gigabit millimeter wave wireless communications for 5G: From fixed access to cellular networks," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 168–178, 2015.

[2] H. Xu, V. Kukshya, and T. S. Rappaport, "Spatial and temporal characteristics of 60-GHz indoor channels," *IEEE Journal on Selected Areas in Communications*, 2002.

[3] S. Singh, F. Ziliotto, U. Madhow, E. Belding, and M. Rodwell, "Blockage and directivity in 60 GHz wireless personal area networks: From cross-layer model to multihop mac design," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 8, pp. 1400–1413, 2009.

[4] S. Collonge, G. Zaharia, and G. E. Zein, "Influence of the human activity on wide-band characteristics of the 60 GHz indoor radio channel," *IEEE Transactions on Wireless Communications*, vol. 3, no. 6, pp. 2396–2406, 2004.

[5] M. Umehira, G. Saito, S. Wada, S. Takeda, T. Miyajima, and K. Kagoshima, "Feasibility of RSSI based access network detection for multi-band WLAN using 2.4/5 GHz and 60 GHz," in *2014 International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pp. 243–248, IEEE, 2014.

[6] Y. Sun, G. Feng, S. Qin, Y.-C. Liang, and T.-S. P. Yum, "The smart handoff policy for millimeter wave heterogeneous cellular networks," *IEEE Transactions on Mobile Computing*, vol. 17, no. 6, pp. 1456–1468, 2017.

[7] S. Zang, W. Bao, P. L. Yeoh, H. Chen, Z. Lin, B. Vucetic, and Y. Li, "Mobility handover optimization in millimeter wave heterogeneous networks," in *2017 17th International Symposium on Communications and Information Technologies (ISCIT)*, pp. 1–6, IEEE, 2017.

[8] Y. Oguma, T. Nishio, K. Yamamoto, and M. Morikura, "Proactive handover based on human blockage prediction using RGB-D cameras for mmWave communications," *IEICE Transactions on Communications*, vol. 99, no. 8, pp. 1734–1744, 2016.

[9] Y. Oguma, T. Nishio, K. Yamamoto, and M. Morikura, "Implementation and evaluation of reactive base station selection for human blockage in mmWave communications," in *2015 21st Asia-Pacific Conference on Communications (APCC)*, pp. 199–203, IEEE, 2015.

[10] A. Mazloum, E. Kfoury, J. Gomez, and J. Crichigno, "A survey on rerouting techniques with P4 programmable data plane switches," *Computer Networks*, 2023.

[11] J. Crichigno, N. Ghani, J. Khoury, W. Shu, M. Wu, "Dynamic routing optimization in WDM networks," in *IEEE 2010 Global Telecommunications Conference (GLOBECOM)*, 2010.

[12] J. Crichigno, W. Shu, M. Wu, "Throughput optimization and traffic engineering in WDM networks considering multiple metrics," in *IEEE 2010 International Conference on Communications (ICC)*, 2010.

[13] E. Kfoury, J. Crichigno, E. Bou-Harb, D. Khoury, and G. Srivastava, "Enabling TCP pacing using programmable data plane switches," in *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, 2019.

[14] M. Mezzavilla, S. Goyal, S. Panwar, S. Rangan, and M. Zorzi, "An MDP model for optimal handover decisions in mmWave cellular networks," in *2016 European Conference on Networks and Communications (EuCNC)*, pp. 100–105, IEEE, 2016.

[15] C. Dhahri and T. Ohtsuki, "Q-learning cell selection for femtocell networks: Single-and multi-user case," in *2012 IEEE Global Communications Conference (GLOBECOM)*, pp. 4975–4980, IEEE, 2012.

[16] H. Tabrizi, G. Farhadi, and J. Cioffi, "Dynamic handoff decision in heterogeneous wireless systems: Q-learning approach," in *2012 IEEE International Conference on Communications (ICC)*, pp. 3217–3222, IEEE, 2012.

[17] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3, pp. 279–292, 1992.

[18] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Reinforcement learning based predictive handover for pedestrian-aware mmWave networks," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops*, pp. 692–697, IEEE, 2018.

[19] K. Song, R. Cai, and D. Liu, "A fast relay selection algorithm over 60 GHz mmWave systems," in *2013 15th IEEE International Conference on Communication Technology*, pp. 676–680, IEEE, 2013.

[20] Y. M. Tsang and A. S. Poon, "Detecting human blockage and device movement in mmWave communication system," in *2011 IEEE Global Telecommunications Conference-GLOBECOM 2011*, pp. 1–6, IEEE, 2011.

[21] E. F. Kfoury, J. Crichigno, and E. Bou-Harb, "An exhaustive survey on P4 programmable data plane switches: Taxonomy, applications, challenges, and future trends," *IEEE Access*, vol. 9, 2021.

[22] Netgear, "WiFi Router AD7200." [Online]. Available: https://tinyurl.com/2p8se37x, Accessed on 3-28-2023.

[23] "ASUS ROG PHONE full specifications." [Online]. Available: https://tinyurl.com/3dpw6ybs, Accessed on 3-28-2023.

[24] Y. Koda, K. Nakashima, K. Yamamoto, T. Nishio, and M. Morikura, "Handover management for mmWave networks with proactive performance prediction using camera images and deep reinforcement learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 2, pp. 802–816, 2019.

[25] T. Wei and X. Zhang, "Pose information assisted 60 GHz networks: Towards seamless coverage and mobility support," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, pp. 42–55, 2017.

[26] Kvalo, "Linux wil6210 Driver." [Online]. Available: https://tinyurl.com/2p8jnwk2, Accessed on 3-28-2023.

[27] "iPerf - the ultimate speed test tool for TCP, UDP and SCTP." [Online]. Available: https://tinyurl.com/47vzrv8k, Accessed on 3-28-2023.

[28] X. Zhu, A. Doufexi, and T. Kocak, "Throughput and coverage performance for IEEE 802.11ad millimeter-wave WPANs," in *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*, pp. 1–5, IEEE, 2011.

[29] N. Moraitis and P. Constantinou, "Indoor channel measurements and characterization at 60 GHz for wireless local area network applications," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 12, pp. 3180–3189, 2004.

[30] T. Nitsche, G. Bielsa, I. Tejado, A. Loch, and J. Widmer, "Boon and bane of 60 GHz networks: Practical insights into beamforming, interference, and frame level operation," in *Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies*, pp. 1–13, 2015.

[31] "Androidblobs." [Online]. Available: https://tinyurl.com/2c8ee485, Accessed on 3-28-2023.