A Case for Temperature-Aware Scheduler for Millimeter-Wave Devices and Networks

Paper #194

***Abstract*—Millimeter-wave is the core technology to enable multi-Gbps throughput and ultra-low latency wireless connectivity. But, the devices need to operate at very high frequency and bandwidth; so, they consume more energy, dissipate more power, and subsequently heat up faster. Device overheating is a common concern of many users, and millimeter-wave (mmWave) would exacerbate the problem. In this work, we first perform a measurement study of mmWave devices' thermal performance. Our measurements reveal that after only 10 s. of data transfer at 1.9 Gbps bit-rate, the mmWave antenna temperature reaches 68°C; it reduces the link throughput by 21%, increases the standard deviation of throughput by 6×, and takes 130 s. to dissipate the heat completely. Besides degrading the user experience, exposure to high device temperature also creates discomfort. Based on the measurement insights, we propose *Aquilo*, a temperature-aware multi-antenna scheduler; it maintains relatively high throughput performance but cools down the devices substantially. Our testbed experiments in both static and mobile conditions show that *Aquilo* reaches a median peak temperature just 0.5 to 2°C above the optimal by sacrificing less than 10% of throughput.**

I. Introduction

The explosive demand for mobile broadband globally has created significant stress on the existing wireless infrastructure. Millimeter-wave (mmWave) has emerged as the core, new technology for the next-generation wireless LAN and cellular standards: IEEE 802.11ad [1]; IEEE 802.11ay [2]; and 5G NR [3], [4]. MmWave systems are the key enabler for use cases that demand multiple Gbps throughput and ultra-low latency connectivity — immersive virtual and mixed reality, tactile internet, telesurgery, control for smart infrastructures, and autonomous vehicles safety [3]–[6]. MmWave systems achieve these performances by operating at a very high frequency and ultra-wide bandwidth, on the order of multiple GHz.

Such a high operational regime, however, brings unique challenges: Compared to the micro-wave devices, like Wi-Fi, the mmWave devices consume more energy, dissipate more power, and subsequently heat up faster. Increase in device temperature not only affects the hardware but also is disconcerting to the users, especially when devices are small, *e.g.*, smartwatches [7], hand-held, *e.g.*, 5G smartphones [8]–[10], body-worn [11], [12], and near to the face and brain, *e.g.*, wireless VR and AR platforms [13]–[16]. Device overheating is a common concern of many users, and mmWave would exacerbate the problem further. Thus, investigating ways to mitigate thermal-inefficiencies in mmWave devices is of vital importance.

Existing research works have extensively characterized mmWave channel, link, network, and applications; however, the thermal characteristics of the mmWave device are relatively less understood. Compared to the micro-wave devices, mmWave devices operate at a very high frequency and ultra-wide bandwidth; thus, each hardware component, *e.g.*, baseband, ADC/DACs, PLLs, mixers, power amplifiers, *etc.*, consumes more energy and dissipate more power. Although mmWave devices are more energy-efficient (consume less energy per bit) than Wi-Fi or LTE [17], [18], higher aggregate energy consumption and power dissipation can heat the mmWave devices quicker. While existing works have looked into the power consumption of the commercial mmWave devices [17], [18], to the best of our knowledge, *none has looked at the thermal characteristics of the mmWave devices*.

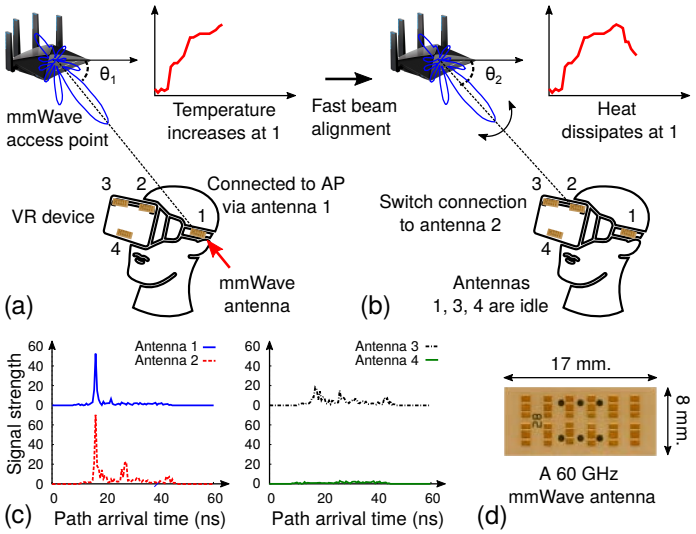

Figure 1: *Core idea behind Aquilo: (a) VR device with 4 mmWave antennas connected to the access point (AP) via antenna 1; data transfer increases its temperature; (b) Scheduler switches to relatively cooler antenna 2 with the best link to the AP; (c) Signal strength of the best beam from the 4 antennas; (d) Dimensions of a 60 GHz mmWave antenna [8].*

To this end, we first characterize the thermal profile of a Commercial-Of-The-Shelf (COTS) 60 GHz mmWave smartphone [10] under various device states and environmental conditions. Our measurements reveal that after only 10 s. of data transfer, at room temperature, with 1.9 Gbps bit-rate, the mmWave antenna temperature reaches up to 68°C; it reduces the average link throughput by more than 21%, increases the standard deviation of throughput by 6×, and takes about 130 s. to dissipate the heat completely. Besides degrading the user experience, an increase in device temperature is disconcerting. More importantly, exposure to a high temperature may not

only create short-term discomfort but also has long-term adverse health effects [19], [20].

Driven by the measurement insights, we propose *Aquilo*[1] — a temperature-aware multi-antenna scheduler that cools down mmWave devices substantially. *Aquilo*'s key idea is intuitive: Before one antenna heats up excessively, its data stream may be switched or distributed to other redundant antennas, allowing it to dissipate the heat. Equipping a mmWave system, like 5G smartphone or VR, with multiple antennas not only is a reasonable system's choice but also is necessary to provide reliable connectivity under channel fluctuations and obstructions [8], [21], [22]. Coordination among these multiple antennas *proactively* can reduce the overall temperature. This idea draws inspiration from the existing *thermal mitigation* techniques in multi-processor architectures [23]–[27]; yet, the challenges are in the variable thermal behavior and variable connectivity of the mmWave antennas. We propose a smart, adaptive multi-antenna scheduling technique that exploits the near-past observation of the thermal profiles, and probe and switch scheme to maintaining relatively stable throughput performance while reducing the overall device temperature.

We have validated *Aquilo* on a 60 GHz mobile mmWave testbed; it consists of a NETGEAR X10 access point (AP) [28] and ASUS ROG smartphones [10]. Both the AP and smartphones are IEEE 802.11ad standard-compliant; also, ASUS ROG is the only commercially available 60 GHz smartphone, currently. Since the COTS smartphone is equipped with one mmWave antenna only, we collect real device throughput and temperature profiles, but use trace-based methods to emulate a multi-antenna device. Our testbed experiments demonstrate that, in comparison to a *throughput-only maximization* scheduling, *Aquilo* can effectively reduce the median peak temperature by 12°C and 9.5°C, under static and mobile conditions, respectively. While these improvements come from sacrificing 9.8% (static) and 8.5% (mobile) of throughput, *Aquilo* is still able to support at least 1.4 Gbps throughput at all times in static and 1.03 Gbps in more than half of the mobile cases. Furthermore, our field trials with traces collected from two applications, FTP and VR, show that *Aquilo* satisfies the minimum throughput requirements while simultaneously achieving the near-optimal device temperatures.

In summary, we have two main contributions:

(1) **Thermal Characterization of 60 GHz mmWave Device**: To the best of our knowledge, we are the first to perform a study of the effects of mmWave device states and link performance on the device's temperature and the effect of temperature on the device's performance.

(2) **Temperature-Aware Multi-Antenna Scheduler**: Based on the insights from our thermal characterization, we propose, design, and validate a temperature-aware multi-antenna scheduler and demonstrate its effectiveness in maintaining the link performance while reducing the temperature substantially.

[1]*Aquilo* was the Roman god of cold north wind and bringer of winter.

II. Background and Related Work

MmWave Devices and Standards: MmWave devices operate at a very high wireless frequency and ultra-wide bandwidth, in the order of multiple GHz. Currently, there are two most popular mmWave standards: 5G NR [3], [4] (frequency ranges are 26.5–29.5 GHz and 37.0–40.0 GHz); and IEEE 802.11ad [1] (57–71 GHz). Specifically, IEEE 802.11ad devices operate on the unlicensed 60 GHz mmWave, use 2.16 GHz bandwidth, and can achieve peak bit-rate up to 7 Gbps. Since mmWave channel suffers from high signal propagation loss, both the standards use phased-array antenna and directional beam for signal strength compensation. Due to the small form factor of mmWave radio-frequency (RF) components and elements, multiple antennas can be integrated into mobile devices [8], [21], [22]; besides, multiple antennas provide reliable connectivity under channel fluctuations and obstructions [29]–[33]. While existing research works have extensively characterized mmWave channel [34]–[51], link [52]–[57], network [58]–[65], and applications [66]–[71], *Aquilo* is the first to characterize the thermal profile of mmWave antennas, and design, evaluate, and demonstrate a temperature-aware multi-antenna scheduling scheme. *Aquilo* is complementary to the existing mmWave systems' research and can benefit from their reliable connectivity and improved performance.

Thermal Mitigation Techniques: *Aquilo* is partly inspired by the thermal mitigation techniques in multi-CPU systems. Several proposed approaches can reduce the temperature significantly: dynamic trigger [23], [72], [73]; hybridized thermal stress-aware adaptation [24]; priority queueing [25]; passive load balancing and active migration [26]; and stochastic techniques [27]. Yet, *Aquilo* faces two challenges that are absent in multi-CPU systems: variable thermal behavior; and highly variable and unpredictable connectivity of the mmWave antennas. While there are existing works in micro-wave systems to minimize the power or energy consumption [74]–[79], to the best of our knowledge, none has looked at minimizing the system-level temperature of active mmWave antennas.

III. Thermal Characterization

In this section, we characterize the impact of device's states and throughput performance on its thermal profiles. *First*, we investigate the thermal profile of mmWave antenna under device's idle and active states. *Second*, we identify performance fluctuations under high antenna temperatures. *Finally*, we study the trade-off between throughput performance and antenna temperature by inducing bit-rate adjustment and periodic idleness. In all cases, we explain the underlying reason for the observed behavior and how it differs from what may be expected. The findings of this section inform our proposal for temperature-aware multi-antenna scheduler.

Setup: For our experiments, we use commercial mmWave smartphones: ASUS ROG [10] and AP: NETGEAR X10

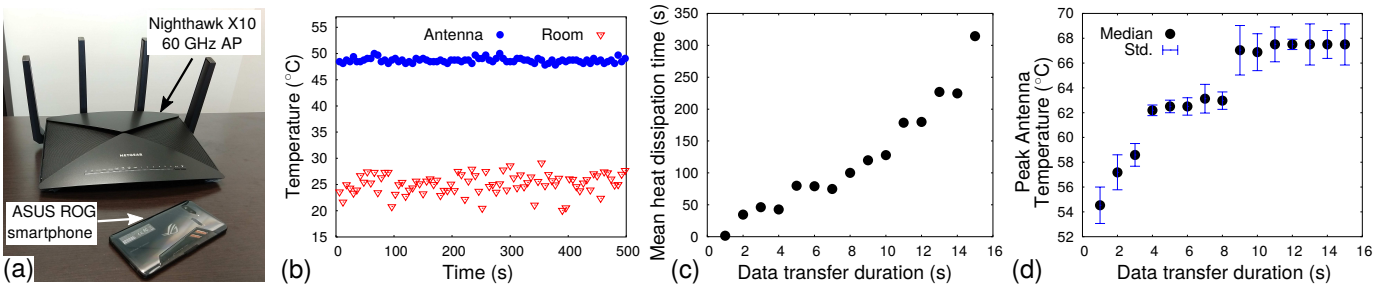

Figure 2: *(a) Millimeter-wave access point (AP) and smartphone. (b) The temperature of the room and the millimeter-wave antenna at idle state. Effect of data transfer duration on: (c) Mean heat dissipation time; and (d) Peak antenna temperature.*

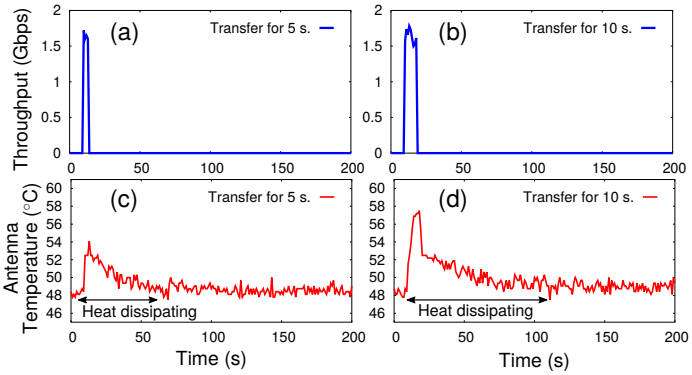

Figure 3: *(a–b) Data transfers for 5 s. and 10 s. (c–d) Antenna temperatures over time and heat dissipation durations.*

[28] (Figure 2[a]). The devices operate at 60 GHz mmWave and are IEEE 802.11ad compliant [1]. They use Qualcomm mmWave chipset [80] and operate on a 2.16 GHz bandwidth, support up to 4.62 Gbps bit-rate, and have a 32-elements phased-array antenna [58], [60]. Furthermore, the devices have embedded, high-resolution temperature sensor that allows us to monitor the antenna's temperature continuously. The smartphones use Android version 8.1 with a Linux kernel, and we can access the temperature data directly from the kernel file: /sys/kernel/debug/ieee80211/phy0/wil6210/temp. We performed experiments in a temperature-controlled indoor office environment with static mmWave channel conditions.

A. Thermal Profile In Idle and Active States

Profile in Idle State: We start by understanding the impact of the device's idle and active states on the thermal profile. We first measure the temperature when the mmWave antenna is idle, *i.e.*, no data communication with the AP. Figure 2(b) shows that the antenna's average temperature is 48.63°C, about 24°C higher than the room temperature. This high idle temperature is due to the very high power consumption during idle listening [18], [78] — the smartphone needs to listen to the incoming mmWave packets and assess the clear channel condition continuously [1]. This also corroborates with the existing measurement that shows the idle listening power consumption in mmWave devices can be up to 1.7 W [18].

Profile in Active States: Active states consume more power, about 2.5 W during data transfer [18], [78]; thus, it increases the antenna temperature further. Nonetheless, we expect that, as soon as the data transfer stops, the antenna would cool down and reach its idle temperature quickly. Unfortunately, heat acts just like stored energy in a capacitor — it takes a long time to dissipate, and more the data transfer, longer the heat dissipation time. To understand this effect, we set up the smartphone 1 m. away from the AP and transfer data at a peak rate for a specific duration; the duration varies between 1 s. and 15 s. In parallel, we measure the antenna temperature.

Figures 3(a–d) plot two examples of antenna temperature for 5 s. and 10 s. of data transfers. They show that a 10 s. of transfer requires about 130 s. to dissipate the heat completely. The dissipation time is measured as the time it takes the antenna to return to its idle temperature. Increase in the transfer duration also increases the dissipation time; our measurements in Figure 2(c) is in agreement. More importantly, *the average dissipation time can be up to 20× higher than the transfer duration*. These transfer durations are lower than a typical application run-time; moreover, certain applications, like wireless AR/VR streaming or gaming, require continuous Gbps data transfer. Besides, higher data transfer duration affects the antenna's peak temperature too; Figure 2(d) shows that the temperature can reach up to 68°C for only a 10 s. data transfer.

B. Performance Fluctuations with High Temperature

High antenna temperature can adversely affect the throughput performance too, due to increased thermal noise and leakage current. To quantify this effect, we perform experiments under the previous set up, but change the AP-smartphone distance to 50 cm. *First*, we transfer data at a peak data rate for 90 s. duration continuously and measure the antenna temperature as it rises up. *Then*, we divide the measurement into several time segments, marking every time index where the temperature increases by approximately 2°C. *Finally*, we measure the average and standard deviation of the throughput for each time segments. We repeat our experiments 85 times and present average of the results. Throughout the experiments, we also measure the Received Signal Strength (RSS) by the AP. The standard deviation of the RSS is 0.47 dB only; in other words, the data is transferred at a very stable channel condition.

Figure 4(a) shows an example RSS and throughput profile and marks the time indices for each segments. Even under the sta-

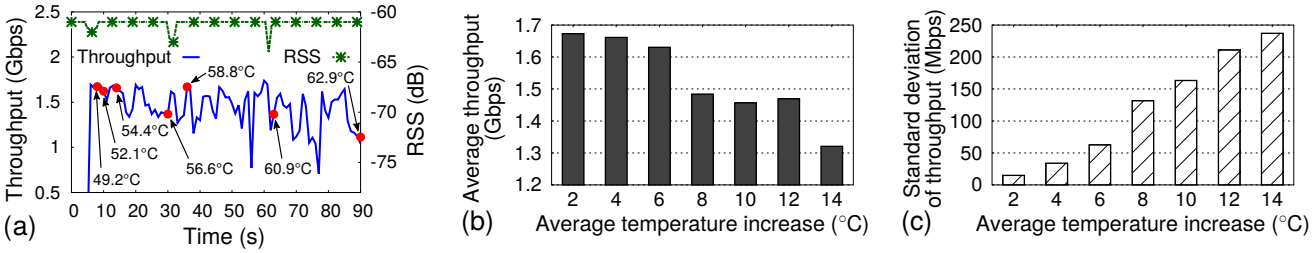

Figure 4: *Link performance is affected by higher antenna temperature: (a) Link RSS is stable, but throughput fluctuates over time; every time index is marked for 2°C temperature increase; For every 2°C increase, we see changes in (b) Average throughput; and (c) Standard deviation of throughput.*

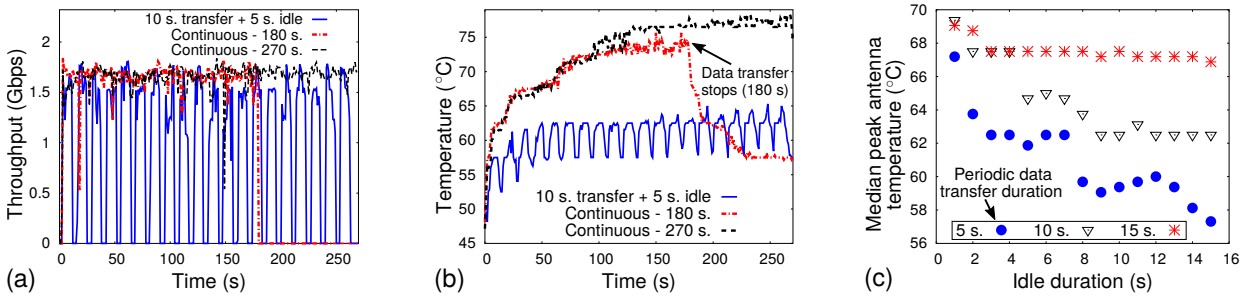

Figure 5: *Under three data transfer modes: (a) Throughput of the millimeter-wave link; and (b) Temperature profiles of the antenna. (c) Effect of various idle durations on antenna's peak temperature.*

ble channel condition, the throughput fluctuates significantly. Figures 4(b–c) plot the average throughput and its standard deviation for every 2°C rise in temperature, upto 14°C maximum increase. Clearly, average throughput degrades as the temperature increases; the loss of average throughput at 14°C increase is 21%. Besides, standard deviation of the throughput increases by almost 6×, even under stable channel condition; it can reach up to 240 Mbps for a 14° rise in temperature. Thus, *higher antenna temperature not only degrades average performance but also increases its variations significantly*. Since both noise and leakage current degrades signal fidelity, we speculate that the device adapts to it by changing the bit-rate. But, commodity devices currently do not allow measuring such fine-grained bit-rate adaptation, thermal noise, or leakage current; so, we leave an extensive analysis of the performance fluctuations under high temperature as future work.

C. Effect of Periodic Idleness and Bit-Rate Adjustment

Inducing Periodic Idleness: A strawman approach to reducing the antenna temperature is to keep it idle in between the data transfers, since *an idle period helps the antenna to cool down*. To understand this effect, we use the previous experimental set up and run experiments for 270 s. with three data transfer modes: continuous for 270 s; periodic for 10 s. followed by idleness for 5 s. (*i.e.*, 66.7% duty cycle); and continuous for 180 s. (66.7% of 270 s.). Figures 5(a–b) show the throughput and temperature profiles. Under continuous 180 s. and 270 s. transfers, the temperatures grow steadily peaking at 75.63°C and 79.38°C, respectively. The temperature rises fast at the beginning; but, when there is a certain difference with the surrounding, the rate decreases due to heat loss, causing marginal increments, and platauing off at the peak. After 180 s., when the transfer stops, the antenna starts cooling down; but, it never reaches to the idle temperature within 270 s. Introducing a periodic idleness of 5 s. after every 10 s. of data transfer reduces the temperature substantially; the peak temperature is below 64°C, for 96% of the time. Figure 5(c) also shows that longer idleness in between transfers can reduce the peak temperature when the transfer durations are short. But, this is ineffective for a longer transfer duration. For example, when the data transfer duration is periodic 15 s., none of the idle durations from 3 to 15 s. can reduce the peak temperature below 66°C.

Inducing Bit-Rate Adjustment: Another approach to reducing the temperature would be to slow down data transfer speed. Intuitively, *the device consumes less power to transfer at a lower bit-rate, thus, heats up slowly*. This is reflected in Figure 6(a); it illustrates three experiments at different mean throughput-level. For the mean throughputs at ∼1.4 Gbps, ∼0.9 Gbps, and ∼0.024 Gbps, the peak temperatures are at 67.5°C, 63°C, and 56°C, respectively. However, Figure 6(b) shows that the temperature cannot be reduced by changing the throughput-level anywhere from 1.6 to 1 Gbps, hardly 0.17°C. Furthermore, for certain devices and applications, *e.g.*, wireless VR/AR, *neither increased idle duration nor reduced bit-rate is affordable since they require stringent throughput and latency guarantees*; but it may be possible to switch the data stream to another antenna with lower temperature.

D. Measurements Summary

In summary, we showed the following properties:

- MmWave device's temperature could be high, even when the device is idle; moreover, a longer data transfer duration

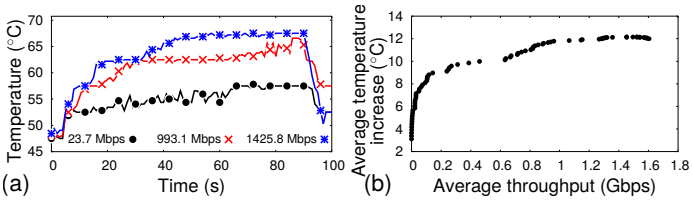

Figure 6: *(a) Temperature profile of 85 s. of data transfer; (b) Average temperature increase for different link throughput.*

not only increases the peak temperature but also takes significantly longer time for heat dissipation — the dissipation time can be up to 20× higher than the transfer duration.

- This high device temperature, in turn, affects the link performance: even at static conditions, it reduces the average link throughput by more than 21% and increases the standard deviation of throughput by 6×.
- A lower bit-rate can reduce the device temperature, only when the link throughput is below 1 Gbps; more importantly, periodic idleness can help reduce antenna's temperature, but only when the data transfer duration is short.

IV. Aquilo Design

Driven by the measurement insights, we propose *Aquilo* — a thermal profile based multi-antenna scheduler to maximize the link performance (*e.g.*, throughput) and minimize the antenna temperature. *Aquilo* enables a mmWave AP and end-user device to select relatively cooler antennas dynamically. While existing works in micro-wave systems may reduce the idle temperature by scheduling sleep periods [1], [74]–[77] or reducing ADC bit-rate or clock-rate [78], [79], to the best of our knowledge, *none has looked at minimizing the system-level temperature of active mmWave antennas.* Unfortunately, the objectives of minimizing temperature and maximizing performance, in a single antenna system, are perennially in conflict; this is because an antenna's temperature rises steadily while transferring data continuously (Section III). Fortunately, upcoming mmWave systems, like 5G NR smartphones and VR/AR devices, are being equipped with multiple antennas as it is not only a reasonable system's choice but also necessary to provide reliable connectivity [8], [21], [22]. *Aquilo* leverages the presense of and coordination among these multiple antennas to reduce the overall system's temperature.

Multi-Antenna Coordination: At a high level, before one antenna heats up excessively, its data stream may be switched to the other antennas, allowing it to dissipate heat. This idea is partly inspired by the *thermal mitigation* techniques in multi-CPU systems [23]–[27], [72], [73]. But, there exist significant differences between CPU's and mmWave antenna's working principle. *First*, while any one of the CPUs can be turned on and expected to work, such an assumption is invalid for the mmWave antennas; this is because channel fluctuations and antenna obstructions may not allow for a link establishment. *Second*, the thermal behavior of the antenna quickly changes depending on the link performance and user's handling of the device, in addition to device insulations and surrounding temperatures. So, multi-antenna coordination techniques will need to consider at least two mmWave issues not present in current thermal mitigation techniques: (1) *variable thermal behavior*; and (2) *highly variable and unpredictable connectivity*.

To address these two issues, we propose to *use an online thermal profile estimation based look-ahead scheduling* among multiple mmWave antennas. The key idea is intuitive. Data transfer from an antenna increases its temperature while idleness decreases it. Furthermore, the increase and decrease rates depend on the amount and duration of transferred data (Section III). Thus, if we could somehow predict an antenna's temperature when they are transferring data at a future point in time, we would be able to schedule a set of antennas such that the peak temperature is minimized and certain performance criterion is met. Unfortunately, the temperature increase and decrease rates are not always deterministic; besides, it's hard to predict mmWave link performance ahead of time because of highly variable connectivity. *Aquilo* leverages near-past observations of the thermal profile for the temperature prediction and a look-ahead schedule and probe scheme for antenna selection. Next, we describe these design components in detail.

A. Thermal Profile Estimation

Issues with a Fixed Thermal Profile Model: A natural way to predict an antenna's future temperature would be to use a fixed temperature rise and fall model. Intuitively, the model parameters depend on the data transfer rate and duration; and, based on several measurements, we could extract the parameters and use them during the run-time. Unfortunately, the model itself varies depending on several factors: device types; insulation materials, locations, and amount; surrounding temperature; and user's handling of the device. To understand the model variations, we measure the thermal profile using the AP-smartphone set up under two environments: indoor office; and home. We set up the smartphone 50 cm. away from the AP and transfer data at a peak rate for 10 s. continuously, ensuring that the channel is stable and the average throughput is around 1.6 Gbps under both the environments. Figures 7(a–b) show the resulting thermal profiles under the two environments. Clearly, they differ significantly in terms of three parameters: peak temperature; heat dissipation time; and noise. Thus, a fixed temperature rise and fall model established from pre-measured dataset will not be generalizable.

Online Profile Estimation: Instead of relying on a fixed model, *Aquilo* leverages online measurements from near-past time, by recording the temperature of the antennas when they are active or idle. But, from our observation in Figures 3 (Section III) and Figures 7(a–b), clearly, we need to model the active and idle states separately; this is because switching an antenna from active to idle state stops increasing and starts decreasing the temperature immediately, creating a stark discontinuity in the thermal profile. Thus, the thermal profile of a mmWave antenna can be modeled as two exponentials: *exponential gain*, $e^{\alpha \cdot t}$, in the active state; and *exponential fall*,

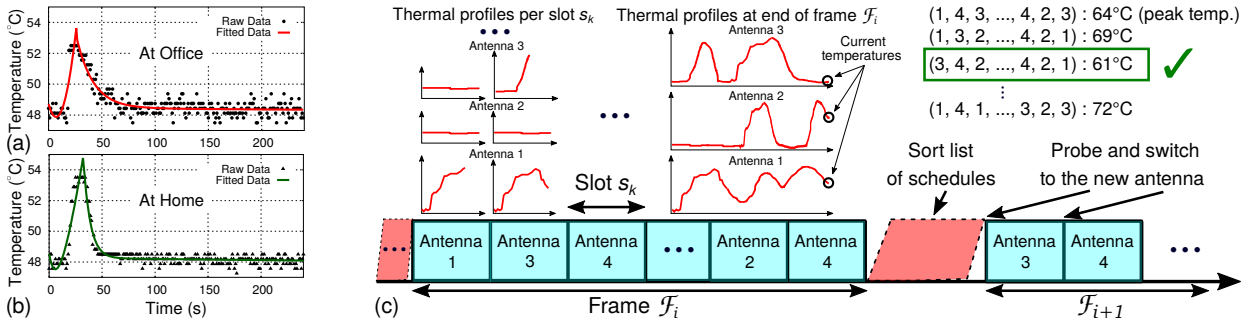

Figure 7: *Data transfer for 10 s., and then, heat dissipation: (a) At office; and (b) At home. (c) Aquilo schedules antennas at the beginning of every frame $\mathcal{F}_i$; then, probes and switches to a new antenna every slot s_k.*

$e^{-\beta \cdot t}$, in the idle state. Furthermore, we can estimate the α and β parameters from the near-past temperature observations. Despite this discontinuity, these separate models serve us better in estimating the thermal profiles accurately; the average root mean square error for model fit across all our measured thermal profiles is only 0.52°C. Figures 7(a–b) also show two such fitting results over the raw measurements. Unfortunately, such raw data is unavailable when the device boots up, or the antenna wakes up for the first time. In such cases, *Aquilo* can randomly select and switch between antennas to bootstrap the thermal profiles. Note that, such random selection and switch happen only once per antenna since when an antenna starts data transfer, we can measure its thermal profile immediately.

B. Look-ahead Schedule, Antenna Probe and Switch

Peak Temperature Based Schedule: *Aquilo* leverages the estimated thermal profiles to find a list of antenna schedules. Intuitively, the profile indicates the likelihood and change of an antenna's temperature when it's assigned for data transfer at a future point in time. So, under each schedule, the antennas will go through a temperature transformation, and thus, would reach a different peak temperature potentially. Therefore, we can pre-compute the peak temperature attained by each schedule and select the one with the lowest peak. Selecting a schedule with lowest peak temperature also ensures that there is no possibility of a significant temperature differential, *i.e.*, one antenna heating up too high while the others are remaining at idle states. Let's consider a simple example to illustrate this point. Assume we have a 3 antenna system; each antenna is initially at idle temperature, 48°C. We would like to schedule antennas every 1 s. interval; therefore, for a total of 3 s., there are 27 (3^3) possible schedules: (1, 1, 1); (1, 1, 2); ...; (1, 2, 3); ...; (3, 3, 3). Furthermore, let's assume that each antenna's temperature increases by 2°C every 1 s. when they are scheduled and transfer data. Hence, we can pre-compute the peak temperature of the schedules: (1, 1, 1) → 54°C; (1, 1, 2) → 52°C; ...; (1, 2, 3) → 50°C; ...; (3, 3, 3) → 54°C. Clearly, schedule (1, 2, 3) has the lowest peak temperature; thus, *Aquilo* will choose it for future data transfer.

Unpredictable Connectivity: The above simple example assumes that there is an equal likelihood of selecting any one of the three antennas. In mmWave, however, the connectivity is highly variable and unpredictable: channel fluctuations and rampant obstructions from the user's hand, body, and various environmental objects may block one or many antennas [40], [41], [51]–[55], [58]. Thus, the antennas may not have a strong enough link or sustain an application's performance requirement. To overcome this challenge, *Aquilo* proposes an antenna probing scheme before selecting and switching to it. *First*, *Aquilo* sorts the list of potential schedules as per the likelihood of the lowest peak temperature and then selects the first choice; but, there is no guarantee on the first antenna's performance in the selection. *Next*, *Aquilo* invokes a fast beam alignment protocol [1], [55], [60] to probe for link strength towards the AP, and using an effective SNR metric [81] converts the strength to throughput performance. *Finally*, *Aquilo* switches to the new mmWave antenna that satisfies the expected performance requirement of the application.

Since the first selection may not have sufficient link strength, *Aquilo* iterates through the list of sorted schedules to eliminate the ones beginning with the antenna without the link. It, then, selects the first antenna from the resulting list, which ensures a good quality link as well as a lower peak temperature. Still, there could be scenarios where no antennas in the selected schedule have any link towards the AP. In such cases, *Aquilo* falls back to the currently active antenna to at least guarantee connectivity. Furthermore, antenna probing and switching take a relatively small amount of time with state-of-the-art beam alignment protocols (less than 0.5 ms. [55], [60]); thus, the latency overhead from *Aquilo* will be very low.

Practical Considerations: Ideally, *Aquilo* should estimate the thermal profiles, sort the list of schedules, and probe and switch to a new mmWave antenna continuously. However, such continuous operations will not only be computationally expensive but also incur high probing and switching overheads. For practical considerations, *Aquilo*, thus, operates in a chunk of discrete-time, which we call a *frame* $\mathcal{F}$. At the beginning of i^{th} frame, $\mathcal{F}_i$, based on the past frame's thermal profile estimations, *Aquilo* predicts the temperature of each of the antennas as if they are scheduled for data transfer in $\mathcal{F}_i$. Based on the predictions, *Aquilo* sorts a list of antenna schedules and selects the one with the lowest peak temperature. Each antenna transfers data for a limited time only, which we call a *slot* s; and, at the beginning of every slot s_k, *Aquilo* probes

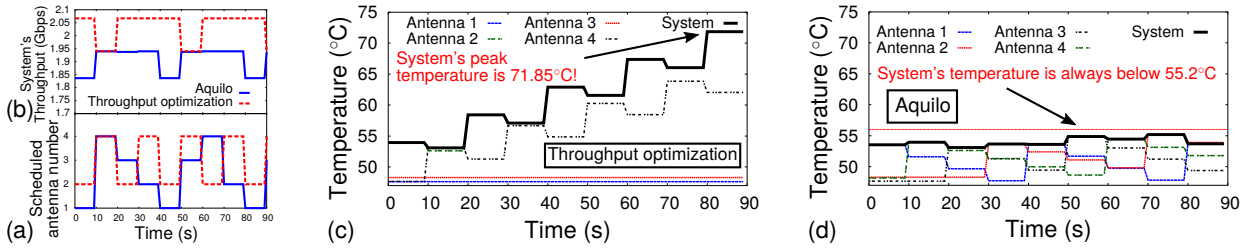

Figure 8: *Example results for a near-static 90 s. of data transfer under Aquilo and throughput optimization. (a) Scheduled antenna numbers; (b) System's throughput; (c–d) Temperature changes over time. Under throughput optimization, peak temperature can reach 71.85°C; but, under Aquilo, the system's temperature never crosses beyond 55.20°C.*

and switches to the new antenna as per the selected schedule. In parallel, *Aquilo* constantly monitors the temperature of both the active and idle antennas, and update a running estimation of their thermal profiles. At the end of $\mathcal{F}_i$, *Aquilo* uses the newly estimated thermal profiles for scheduling in the next frame, $\mathcal{F}_{i+1}$; and the above process repeats. Figure 7(c) shows an illustrative run-time example of *Aquilo*.

A key challenge is to identify the suitable lengths for the slot and frame. Intuitively, *Aquilo* will incur high probing and frequent switching overheads if the slot length is too small. On the other hand, the scheduled antenna will likely change performance within one slot if the length is too large. Besides, slot length and number of slots determine the frame length (slot length $\times$ number of slots). A larger frame length means better thermal profiles estimation; but that profile may not be usable for the next frame since the estimation may become stale. This staleness is due to the temporal variations of different factors, such as the surrounding temperature and the location of the user's hand or body. Furthermore, if the frame length is too large with many slots, *Aquilo* will incur a high computational overhead for thermal profile estimation. For example, if there are 4 mmWave antennas and 5 slots per frame, then the number of possible schedules is 1,024; with only 8 slots per frame, it increases to 65,536.

Non-Adjacency Criteria: One way *Aquilo* reduces this huge list of schedules is by leveraging a simple heuristic: Avoid scheduling the same antenna in back-to-back time slots. We call it a *Non-Adjacency Criteria*. Intuitively, the peak temperature attained by a schedule where the same antenna is never scheduled back-to-back is always lower than any other schedule; this is because an antenna starts to cool down immediately after it stops transferring data (Section III). This heuristic can reduce the number of schedules exponentially. In the previous example, the number of schedules with *Non-Adjacency Criteria* is 8,748, an almost 8$\times$ reduction from 65,536. In Section VI, we have also verified the performance of this heuristic empirically in reducing the peak temperature.

Example Schedule and Performance from *Aquilo*: Figures 8(a–d) show example output of antenna schedule, system's throughput, and system's peak temperature under *Aquilo*, and contrast the results with a throughput maximization scheme. We set up our smartphone at 1 m. away from the AP and transfer data at a peak rate continuously for 90 s.; the channel remains in a near-static condition. We simulated a 4 antenna mmWave system with slot length 10 s. and frame length 90 s. For visual clarity, we averaged the system's throughput and temperature every 10 s. window. Figures 8(a–b) show the scheduled antenna numbers and the corresponding throughput under the two schemes. Clearly, throughput optimization achieves better throughput; it's about 120 Mbps higher than *Aquilo* on average (*i.e.*, *Aquilo* suffers $\sim$ 6% loss). However, the throughput optimization scheme suffers from a very high system's temperature since it only selects two out of four antennas with the best throughput; its peak temperature reaches 71.85°C. *Aquilo*, on the other hand, selects the antenna for data transfer to minimize the system's temperature. The peak temperature never crosses beyond 55.20°C — *i.e.*, 16.65°C temperature reduction sacrificing only 6% average throughput.

Algorithm for Multi-Antenna Scheduling: Algorithm 1 formally describes the antenna scheduling process in *Aquilo*. At the beginning of every frame $\mathcal{F}_i$, *Aquilo* gathers the thermal profile for all antennas scheduled in frame $\mathcal{F}_{i-1}$; hence, for each schedules s_j, it can pre-compute the peak temperature and create a list of sorted schedules, S. *Aquilo*, then probes the first antenna from the first schedule in the sorted list. Since under variable connectivity, this antenna may fail to establish a link, *Aquilo* continuously probes the other antennas in the first schedule, removes the unusable antennas, and updates the list of schedules. Once *Aquilo* is able to establish a link, it starts the data transfer, adapting its beam direction and bit-rate, and updating its thermal profiles. Finally, after *Aquilo* completes every slot scheduled in the current frame, it uses the new thermal profile estimation in the next frame, $\mathcal{F}_{i+1}$.

C. Latency for Antenna Probing and Switching

Since at the beginning of every slot *Aquilo* probes the selected antenna, it may incur additional latency; this is because, during the antenna probe, no useful data transfer occurs. For a small-sized antenna, with 64 beams, this latency is 2.02 ms. in IEEE 802.11ad [1], [55]; for a large-sized antenna, this latency can be on the order of 100s of ms [52], [53]. In that case, we can use state-of-the-art fast beam alignment protocols to keep the latency below sub-ms. [55], [60]. Nonetheless, *Aquilo* minimizes the overall temperature by switching to a new mmWave antenna, and thus, the number of switches and total latency may be higher than other "temperature-

Algorithm 1 *Antenna Scheduling*

For every frame, $\mathcal{F}_i$; Input: Thermal profiles $Th_k^{(i-1)}$ for all antennas k in $\mathcal{F}_{i-1}$; Output: New thermal profile Th_k^i;
For every schedules, $s_j = \{a_1, a_2, a_3, \ldots, a_\eta\}\ \forall j$
  $PT_{s_j} \leftarrow max\{PT(a_1), \ldots, PT(a_\eta)\}$; *PT*: Peak Temp.;
endFor
Sorted list of schedules: $S = \{s_1, s_2, s_3, \ldots, s_N\}$, *s.t.*, $PT_{s_1} \leq PT_{s_2} \leq \ldots \leq PT_{s_N}$;
Initialize the current schedule: α_i = {};
For each slot, $k \in \{1, \ldots, \eta\}$
  Scheduled antenna, *SA* $\leftarrow s_1(a_k)$; *link* $\leftarrow$ *probe(SA)*;
  while *link* is *false*
    Remove all s_j which starts with antenna *SA*; $S \leftarrow S \setminus s_j, \quad \forall s_j, \quad s.t., \quad SA = s_j(a_k)$;
    SA $\leftarrow s_1(a_k)$; *link* $\leftarrow$ *probe(SA)*;
  endwhile
  $\alpha_i \leftarrow \{\alpha_i | SA\}$; *switch(SA)*;
  Update thermal profile for *SA*: Th_{SA}^i
endFor

unaware" schemes. However, the slot length in *Aquilo* is much longer than beam alignment latency; therefore, the relative overhead is very low. Next-generation mmWave standards also use multiple RF chains so that many antennas can be active simultaneously and operate independently [31]–[33], [82]. (An RF chain consists of amplifiers, modulators, filters, PLLs, ADC/DACs, etc., and processes the wireless signal.) In such cases, *Aquilo* can continue data transfer with the active antenna, while simultaneously probing and preparing to switch to another antenna, incurring no additional latency.

D. Integrating Aquilo with IEEE 802.11ad and 5G NR

Aquilo can be integrated with IEEE 802.11ad/ay or 5G NR COTS devices seamlessly. At a high level, *Aquilo*'s slots and frames span multiple beacon intervals in IEEE 802.11ad (100 ms.) or multiple radio frames in 5G NR (10 ms.). For example, if the slot and frame lengths are 1 s. and 5 s., then one slot and frame span 10 and 50 beacon intervals, respectively, in IEEE 802.11ad. Throughout a beacon interval or a radio frame duration, *Aquilo* can follow the standard protocols to aligning the beam directions and transferring data with the active antenna. At the beginning of each slot, instead of initiating a beam alignment from the current antenna per the standard, *Aquilo* initiates the probing and switches to the appropriate antenna as per its schedule. Since this probe and switch happens after every slot, that spans several beacon intervals, the relative cost is very low. Besides, in the future, *Aquilo* can be integrated into the standard devices to provide *device temperature as a Quality of Service*, by enabling flexible guarantees on the peak temperature of the mmWave devices.

V. Implementation

We implement and evaluate *Aquilo* by collecting throughput and temperature measurements from our COTS testbed (Figure 2[a]). The mmWave antenna on both the AP and smartphones can generate up to 64 transmit and receive beams in 3D; but, throughout our experiments, we fix onto a single beam direction that has the strongest link. Both the AP and smartphones can measure the wireless bit-rate, link throughput, and temperature of its mmWave antenna every 1 s. Throughout our evaluation, we consider IEEE 802.11ad [1] as the underlying standard, and follow its beacon structures, beam alignment process, and antenna switching overheads. We collect measurements for various data transfer periods ranging from 5 to 100 s. Between two measurements, the set up was allowed to cool down to its idle temperature so that all measurements have a common baseline. To expedite the cooling process, we also used an external USB fan.

Since the COTS devices currently do not have multiple mmWave antennas, we run trace-driven simulations. *First*, for each experiment, we collect measurements by fixing the smartphone's orientation. *Then*, we rotate it to 4 different orientations *w.r.t.* the AP to create antenna's different positions. *Finally*, we combine the measurements to emulate a setup with 4 mmWave antennas. However, a challenge with this emulation is that it does not capture the effects of the user's device handling, where she may obstruct one or many mmWave antennas occasionally. To simulate these effects, we introduce random blockage as a probability of antenna producing zero throughputs; furthermore, we vary this probability to simulate the different intensity of obstructions.

VI. Evaluation

In this section, we evaluate *Aquilo*'s ability to identify the temperature-optimal antenna scheduling. We consider *Aquilo*'s performance along two dimensions: system's throughput; and peak temperature. We will show the following in our evaluation. (1) *Aquilo* reaches a median peak temperature just 1°C above the best case while sacrificing 9.8% of throughput under various static conditions. (2) Under various degrees of obstructions, *Aquilo* can approximate the temperature-optimal scheme, with 1.1 to 5.4°C differences, while maintaining above 1.77 Gbps in more than half of the cases. (3) While *Aquilo* needs to trade-off temperature-optimal antennas under higher link throughput requirements, it still outperforms simple heuristic-based scheme by 3.9 to 7.4°C. (4) A larger frame length has poorer performance and higher computational burden, and a smaller slot length has higher probing and switching overheads; in practice, 1 s. slot and 10 s. frames perform well. (5) Under mobility, *Aquilo* shows near-optimal thermal performance, with just 0.5°C median difference from the optimal, even when the link demands 1 Gbps throughput. (6) Finally, for two field trial applications, FTP and VR, *Aquilo* provides a similar quality of experience as the best throughput scheme while reaching near-optimal temperature.

A. Compared Schemes

We compare *Aquilo* with the following four schemes:

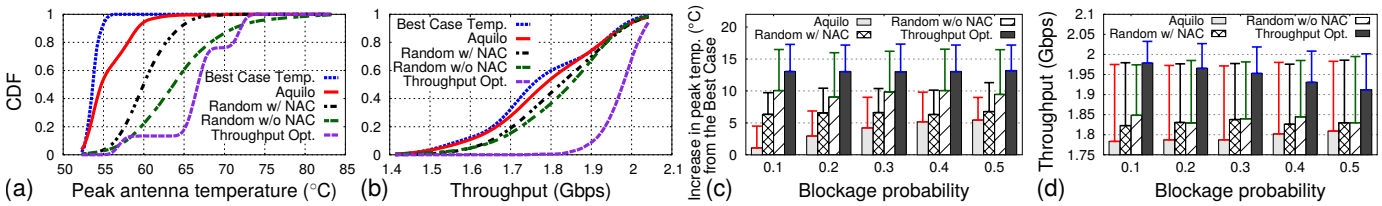

Figure 9: *Empirical CDF results from different schemes across various static conditions: (a) Peak antenna temperature; and (b) System's throughput. Effect of increasing environmental blockage on the performance: (c) Increase in peak antenna temperatures from the best case; and (d) System's throughput. The error bar shows 90th percentile value.*

(1) **Best Case Temperature**: An "oracle" scheme that finds out the minimum temperature of the system under an experimental condition, considering all possible blockage conditions, temperature changes, and antenna performances ahead of time but after the fact. Although an impractical scheme, this estimates the lower bound on the peak temperature.

(2) **Random Scheduling with Non-Adjacency Criteria**: The simplest way to schedule is to *randomly* select an antenna, ensure that it meets the performance requirement, and start data transfer. Since switching an antenna from active to idle state reduces its temperature immediately, we consider a random selection, but ensure that the same antenna is never assigned in adjacent time slots. It allows shuffling between antennas so that the peak temperature does not grow steadily.

(3) **Random Scheduling**: This scheme selects antennas randomly, but without the non-adjacency criteria, and ensure they meet the application's performance requirement.

(4) **Throughput Optimization**: This approach tries to maximize only the throughput, without the knowledge of temperatures. We select the highest throughput schedule from the list, and it determines the upper bound on the throughput.

B. Microbenchmark Results

Performance under Static Conditions: We first measure *Aquilo*'s effectiveness under static conditions. We choose 200 static links, each with 100 s. duration, and estimate the best case peak temperature. To estimate the ground truth of maximum throughput, we also run the throughput optimization scheme. In addition, we run the random scheduling with and without the *Non-Adjacency Criteria* (NAC). While pure random scheduling captures the average case temperature behavior, random scheduling with NAC evaluates a simple thermal mitigation technique: Allow periodic idleness in an antenna by avoiding adjacent time slots. Finally, we compare the throughput and peak temperature results from *Aquilo*.

Figures 9(a–b) show the empirical Cumulative Distribution Function (CDF) of the system's peak temperature and throughput. The median of best case peak temperatures is 53.64°C. The throughput optimization shows the worst case temperature performance; its median peak temperature is beyond 67°C, even worse than random scheduling. This is because it only maximizes the throughput and is "temperature-unaware." Furthermore, a simple heuristic of NAC effectively improves the median peak temperature by more than 3.7°C from the pure random selection. *Aquilo* outperforms all the random schedulings and throughput optimization schemes in terms of temperature; its median peak temperature is just about 1°C above the best case. Compared to the throughput optimization, *Aquilo* suffers from around 200 Mbps throughput loss ($\sim$ 9.8%); but, in more than half of the cases, *Aquilo* reduces the peak temperature by approximately 12°C.

Effect of Environmental Dynamism: Next, we evaluate *Aquilo*'s performance under environmental dynamism that results in varying degrees of antenna blockages. A dynamic environment is the one with moving people, objects around mmWave link, and user's handling of the mmWave device, like orientation changes, partial or complete obstructions of one or more mmWave antennas, *etc.* We have modeled these events as the varying probability of antenna blockage: For example, if the probability is 0.2, then an antenna can successfully establish a link towards the AP only 80% of the time. Figures 9(c–d) show the peak temperature and throughput results with increasing environmental dynamism. Clearly, the temperature performance of *Aquilo* degrades with increasing blockage probability; under 0.5 probability, the median peak temperature is more than 5°C above the optimal. This is expected: As the blockage probability increases, *Aquilo* has limited choices and flexibilities in scheduling antennas with better thermal mitigation, and thus, resort to whatever available for continued operation. While this is a significant drop in the performance, such high blockage conditions are rare occurrences in practice, and will mostly be limited to a very short duration.

Performance under Different Throughput Requirement: Applications like wireless AR/VR have stringent throughput requirements; to support them, *Aquilo* may need to trade-off temperature-optimal antennas for throughput performance. Specifically, Algorithm 1 will be able to schedule only those antennas that guarantee performance at or above the requirements. Thus, intuitively, a higher minimum throughput requirement will degrade the thermal mitigation effectiveness. Figure 10(a) illustrates this point; at higher throughput requirement, *Aquilo*'s performance deviates significantly from the best case and moves closer to random scheduling with NAC.

Effect of Different Frame Lengths: We now evaluate *Aquilo*'s performance in selecting optimal schedules under different frame lengths, but fixed slot lengths. Figure 10(b) shows the difference in median peak temperatures between

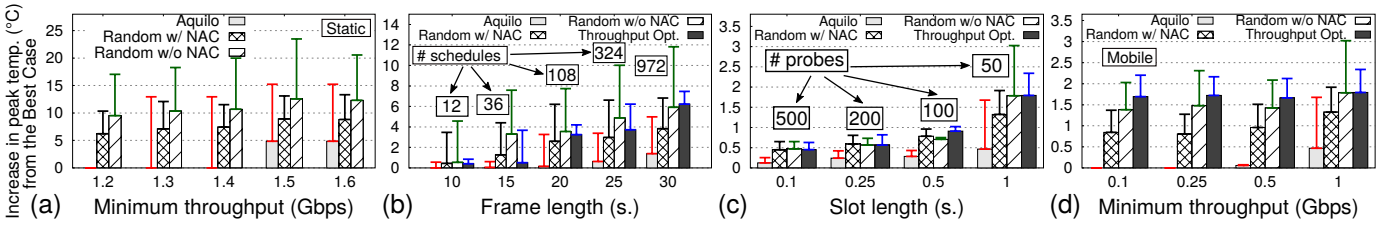

Figure 10: *Increase in peak temperature from the best case. Under static cases, effects of: (a) Minimum throughput requirement; and (b) Frame lengths on peak temperature and # of schedules. Under mobile cases: (c) Effect of slot lengths, on peak temperature and # of probes, when min. throughput is 1 Gbps; and (d) Effect of minimum throughput, when slot length is 1 s.*

the best case and *Aquilo*. We have three observations. *First*, the effectiveness of *Aquilo* drops with higher frame length; this is because *Aquilo* uses past frame's thermal profile estimation to schedule antennas in the next frame, which is more likely to be ineffective if the frame lengths are too large. *Second*, the number of schedules increases with longer frame duration (shown with boxed numbers in Figure 10[b]), and thus, in practice, *Aquilo* will have higher computational overhead. *Finally*, irrespective of the frame length, *Aquilo* still reaches median peak temperature 2°C above the optimal.

Performance under Mobility: We now evaluate *Aquilo*'s performance under mobility. We collect the trace data for various mobility durations between 10 to 50 s., and measure the difference in peak temperatures between the best case and *Aquilo*. We consider two factors in our mobility evaluation: variable slot length; and variable minimum throughput requirement. Figure 10(c) shows that the effectiveness of *Aquilo* increases with smaller slot lengths. This is intuitive: A smaller slot length, such as 0.1 s., allows fast switching between antennas and more frequent schedule updates. However, the total number of probings will be very high (shown with boxed numbers in Figure 10[c]). For example, over 50 seconds, with a slot length of 0.1 s., the number of probings will be 500. Such high overhead may not be worth the extra gain in temperature performance. Nonetheless, a slot length of 1 s. enables reasonable thermal mitigation; and, the median peak temperature is only $\sim$ 0.5°C above the optimal.

Figure 10(d) shows the effect of the minimum throughput requirement of mobile links on *Aquilo*. *Aquilo* achieves near-optimal performance when the requirement is at or below 0.5 Gbps; however, the performance deviates from the optimal, under more stringent 1 Gbps requirement. The performance is also relatively poor compared to the static cases (Figure 10[a]). This is expected: Under mobility, *Aquilo* needs to trade-off temperature-optimal antennas for performance more often than static cases. Still, the median peak temperature is only 0.5°C above the best case, even with 1 Gbps throughput requirement.

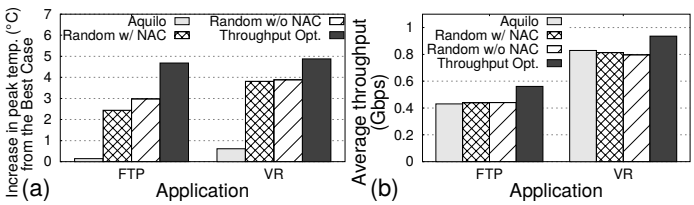

Figure 11: *Performance in field trial applications: (a) System's peak temperature; (b) Application's throughput.*

C. Field Trials

Finally, we evaluate *Aquilo* on real applications; we use previous mobility set up and collect temperature and throughput trace data while running two applications: FTP; and Virtual Reality (VR) gaming. *First*, we set up an FTP server on the AP, and the smartphone downloads a 2 GB file. For each of the downloads, we find the system's average throughput and peak temperature under different schemes. *Then*, we set up a gaming server on a PC [83] that streams real-time data (video, voice, control) over the smartphone via the 60 GHz mmWave link; we also used a Google cardboard [84] to set up a smartphone VR. However, all our field trials are limited to 1 Gbps maximum throughput, since the Ethernet port on the PC does not allow for more. Figures 11(a–b) show the average throughput and peak temperature performance. For both the applications, *Aquilo* maintains the required throughput performance while reducing median peak temperature within 0.75°C above the best case. In summary, *Aquilo provides a similar quality of experience as the best throughput scheme, while simultaneously reaching near-optimal device temperature*.

VII. Conclusion

In this paper, we present the first-of-a-kind study on mmWave thermal characterization; the study reveals new challenges and opportunities to keep IEEE 802.11ad and 5G NR devices cool. Based on the measurement insights, we propose *Aquilo*, the first temperature-aware multi-antenna scheduler at mmWave. We use testbed experiments from COTS mmWave devices to demonstrate *Aquilo*'s effectiveness in maintaining link performance while reducing temperature substantially. In the future, we plan to continue investigating mmWave performance fluctuations under high device temperature with more fine-grained measurements, and characterizing thermal performance under different applications and use cases, both indoor and outdoor; besides, we plan to design, implement, and evaluate a real-time *Aquilo* on commercial multi-antenna mmWave devices. Overall, we believe, our research on mmWave device cooling helps allay concerns in some quarters about health effects of 5G and accelerate its deployment broadly.

REFERENCES

[1] IEEE Standards Association, "IEEE Standards 802.11ad-2012, Amendment 3: Enhancements for Very High Throughput in the 60 GHz Band," goo.gl/s7HzC2, 2012.

[2] IEEE P802.11 - Task Group ay, "Status of Project IEEE 802.11ay," http://www.ieee802.org/11/Reports/tgay_update.htm, 2020.

[3] 3GPP: A Global Initiative, "The Mobile Broadband Standard: Release 15," 2017. [Online]. Available: http://www.3gpp.org/release-15

[4] ——, "The Mobile Broadband Standard: Release 16," 2019. [Online]. Available: http://www.3gpp.org/release-16

[5] Erik Dahlman and Stefan Parkvall and Johan Skold, *5G NR: The Next Generation Wireless Access Technology*. Elsevier, 2018.

[6] Qualcomm Incorporated, "5G mmWave: The Next Frontier in Mobile Broadband." 2018. [Online]. Available: https://www.qualcomm.com/invention/5g/5g-nr/mmwave

[7] Google LLC, "Google Soli," 2019. [Online]. Available: https://atap.google.com/soli/

[8] Qualcomm Incorporated, "Qualcomm Announces 5G NR mmWave Prototype to Accelerate Mobile Deployments for Smartphones," 2017. [Online]. Available: https://www.qualcomm.com/news/releases/2017/09/11/qualcomm-announces-5g-nr-mmwave-prototype-accelerate-mobile-deployments

[9] ——, "Mobilizing mmWave with 5G NR," 2020. [Online]. Available: https://www.qualcomm.com/invention/5g/5g-nr/mmwave

[10] AsusTek Computer Inc., "ASUS ROG Phone," 2018. [Online]. Available: https://www.asus.com/us/Phone/ROG-Phone/

[11] Apple Inc., "United States Patent Application: Electronic Devices Having Millimeter Wave Ranging Capabilities," 2019. [Online]. Available: https://bit.ly/2k0rsSr

[12] Patently Apple, "Apple Files Patents for Future iPhones with Gas Sensors & 5G Millimeter Wave Antennas for Apple Watch ," 2019. [Online]. Available: https://www.patentlyapple.com/patently-apple/2019/08/apple-files-patents-for-future-iphones-with-gas-sensors-5g-millimeter-wave-antennas-for-apple-watch.html

[13] TPCAST U.S., Inc., "TPCAST: Unleash the VR World," 2019. [Online]. Available: https://www.tpcastvr.com/

[14] HTC, "HTC VIVE PRO," 2019. [Online]. Available: https://www.vive.com/us/product/vive-pro/

[15] VentureBeat, "SONY VR Patent Points to a Pricier But Wireless PSVR2," 2019. [Online]. Available: https://venturebeat.com/2019/03/18/sony-vr-patent-points-to-a-pricier-but-wireless-psvr2/

[16] Intel Corporation, "Intel Wireless Gigabit for VR," 2019. [Online]. Available: https://www.intel.com/content/www/us/en/products/docs/wireless-products/wigig-overview.html

[17] Swetank Kumar Saha and Darshan Godabanahal Malleshappa and Avinash Palamanda and Viral Vijay Vira and Anuj Garg and Dimitrios Koutsonikolas, "60 GHz Indoor WLANs: Insights into Performance and Power Consumption," *Journal of Wireless Networks*, vol. 24, no. 7, 2018.

[18] Swetank Kumar Saha and Tariq Siddiqui and Dimitrios Koutsonikolas and Adrian Loch and Joerg Widmer and Ramalingam Sridhar, "A Detailed Look into Power Consumption of Commodity 60 GHz Devices," in *IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2017.

[19] A. Keykhosravi and M. Neamatshahi and R. Mahmoodi and E. Navipour, "Radiation Effects of Mobile Phones and Tablets on the Skin: A Systematic Review," *Advances in Medicine*, 2018.

[20] Om P. Gandhi and Abbas Riazi, "Absorption of Millimeter Waves by Human Beings and Its Biological Implications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 34, no. 2, 1986.

[21] Qualcomm Incorporated, "Breaking the Wireless Barriers to Mobilize 5G NR mmWave," 2019. [Online]. Available: https://www.qualcomm.com/media/documents/files/5g-nr-mmwave-deployment-strategy-presentation.pdf

[22] ——, "5G mmWave Radio Design for Mobile," 2017. [Online]. Available: shorturl.at/bfiFG

[23] D. Brooks and M. Martonosi, "Dynamic Thermal Management for High-Performance Microprocessors," in *International Symposium of High-Performance Computer Architecture (HPCA)*, 2001.

[24] K. Skadron, "Hybrid Architectural Dynamic Thermal Management," in *IEEE Proceedings Design, Automation and Test in Europe Conference and Exhibition*, 2004.

[25] A. Kumar and L. Shang and L-S. Peh and N. K. Jha, "System-Level Dynamic Thermal Management for High-Performance Microprocessors," *IEEE Transactions of Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, 2008.

[26] A. Merkel and F. Bellosa and A. Weissel, "Event-Driven Thermal Management in SMP Systems," in *Workshop on Temperature-Aware Computer Systems*, 2005.

[27] H. Jung and M. Pedram, "Stochastic Dynamic Thermal Management: A Markovian Decision-Based Approach," in *International Conference of Computer Design*, 2007.

[28] NETGEAR, Inc., "Nighthawk X10 Smart WiFi Router," 2017. [Online]. Available: https://www.netgear.com/landings/ad7200/

[29] R. Zhao, T. Woodford, T. Wei, Q. Kun, and X. Zhang, "M-Cube: A Millimeter-Wave Massive MIMO Software Radio," in *Proc. of ACM MobiCom*, 2020.

[30] S. Wang, J. Huang, X. Zhang, H. Kim, and S. Dey, "X-Array: Approximating Omnidirectional Millimeter-Wave Coverage Using an Array of Phased Arrays," in *Proc. of ACM MobiCom*, 2020.

[31] S. Sun, T. S. Rappaport, and M. Shafi, "Hybrid Beamforming for 5G Millimeter-Wave Multi-Cell Networks," in *IEEE Conference on ComputerCommunications Workshop*, 2018.

[32] J. Zhang, X. Yu, and K. B. Letaief, "Hybrid Beamforming for 5G and Beyond Millimeter-Wave Systems: A Holistic View," vol. 1, 2019.

[33] R. W. H. Jr., N. Gonzalez-Prelcic, S. Rangan, W. Roh, and A. Sayeed, " An Overview of Signal Processing Techniques for Millimeter Wave MIMO Systems," vol. 10, no. 3, 2016.

[34] M. K. Samimi and T. S. Rappaport, "3-D Statistical Channel Model for Millimeter-Wave Outdoor Mobile Broadband Communications," in *IEEE International Conference on Communications (ICC)*, 2015.

[35] T. S. Rappaport, E. Ben-Dor, J. N. Murdock, and Y. Qiao, "38 GHz and 60 GHz Angle-Dependent Propagation for Cellular and Peer-to-Peer Wireless Communications," in *IEEE ICC*, 2012.

[36] T. S. Rappaport, F. Gutierrez, E. Ben-Dor, J. N. Murdock, Y. Qiao, and J. I. Tamir, "Broadband Millimeter-Wave Propagation Measurements and Models Using Adaptive-Beam Antennas for Outdoor Urban Cellular Communications," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 4, 2013.

[37] T. Zwick, T. J. Beukema, and H. Nam, "Wideband Channel Sounder With Measurements and Model for the 60 GHz Indoor Radio Channel," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 4, 2005.

[38] H. Xu, T. S. Rappaport, R. J. Boyle, and J. H. Schaffner, "Measurements and Models for 38-GHz Point-to-Multipoint Radiowave Propagation," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, 2000.

[39] C. Anderson and T. Rappaport, "In-Building Wideband Partition Loss Measurements at 2.5 and 60 GHz," *IEEE Transactions on Wireless Communications*, vol. 3, no. 3, 2004.

[40] 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, 2004.

[41] H. Xu, V. Kukshya, and T. S. Rappaport, "Spatial and Temporal Characteristics of 60-GHz Indoor Channels," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 3, 2002.

[42] C. R. Anderson, T. S. Rappaport, K. Bae, A. Verstak, N. Ramakrishnan, W. H. Tranter, C. A. Shaffer, and L. T. Watson, "In-Building Wideband Multipath Characteristics at 2.5 & 60 GHz," in *IEEE Vehicular Technology Conference (VTC)*, 2002.

[43] N. Hakem, G. Delisle, and Y. Coulibaly, "Radio-Wave Propagation into an Underground mine environment at 2.4 GHz, 5.8 GHz and 60 GHz," in *European Conference on Antennas and Propagation (EuCAP)*, 2014.

[44] T. A. Thomas, H. C. Nguyen, G. R. M. Jr., and T. S. Rappaport, "3D mmWave Channel Model Proposal," in *IEEE Vehicular Technology Conference*, 2014.

[45] M. Peter, W. Keusgen, and R. Felbecker, "Measurement and Ray-tracing Simulation of the 60 GHz Indoor Broadband Channel: Model Accuracy and Parameterization," in *European Conference on Antennas and Propagation*, 2007.

[46] V. Degli-Esposti, F. Fuschini, E. M. Vitucci, M. Barbiroli, M. Zoli, L. Tian, X. Yin, D. A. Dupleich, R. Muller, C. Schneider, and R. S. Thoma, "Ray-Tracing-Based mm-Wave Beamforming Assessment," *IEEE Access*, vol. 2, no. 1, 2014.

[47] B. Neekzad, K. Sayrafian-Pour, J. Perez, and J. S. Baras, "Comparison of Ray Tracing Simulations and Millimeter Wave Channel Sounding

Measurements," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2007.
[48] P. Marinier, G. Y. Delisle, and L. Talbi, "A Coverage Prediction Technique for Indoor Wireless Millimeter Waves System," *Wireless Personal Communications*, vol. 3, no. 3, 1996.
[49] Y. Azar, G. N. Wong, K. Wang, R. Mayzus, J. K. Schulz, H. Zhao, F. Gutierrez, D. Hwang, and T. S. Rappaport, "28 GHz Propagation Measurements for Outdoor Cellular Communications using Steerable Beam Antennas in New York city," in *2013 IEEE International Conference on Communications (ICC)*, 2013.
[50] T. Rappaport, F. Gutierrez, E. Ben-Dor, J. Murdock, Y. Qiao, and J. Tamir, "Broadband Millimeter-Wave Propagation Measurements and Models Using Adaptive-Beam Antennas for Outdoor Urban Cellular Communications," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 4, 2013.
[51] P. F. M. Smulders, "Statistical Characterization of 60-GHz Indoor Radio Channels," *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 10, 2009.
[52] Y. Zhu, Z. Zhang, Z. Marzi, C. Nelson, U. Madhow, B. Y. Zhao, and H. Zheng, "Demystifying 60GHz Outdoor Picocells," in *Proc. of ACM MobiCom*, 2014.
[53] S. Sur, V. Venkateswaran, X. Zhang, and P. Ramanathan, "60 GHz Indoor Networking through Flexible Beams: A Link-Level Profiling," in *Proc. of ACM SIGMETRICS*, 2015.
[54] S. Sur, X. Zhang, P. Ramanathan, and R. Chandra, "BeamSpy: Enabling Robust 60 GHz Links Under Blockage," in *Proc. of USENIX NSDI*, 2016.
[55] H. Hassanieh, O. Abari, M. Rodriguez, M. Abdelghany, D. Katabi, and P. Indyk, "Fast Millimeter Wave Beam Alignment," in *Proc. of ACM SIGCOMM*, 2018.
[56] B. Li, Z. Zhou, W. Zou, X. Sun, and G. Du, "On the Efficient Beam-Forming Training for 60GHz Wireless Personal Area Networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 2, 2013.
[57] 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 *ACM CoNEXT*, 2015.
[58] S. Sur, I. Pefkianakis, X. Zhang, and K.-H. Kim, "WiFi-Assisted 60 GHz Wireless Networks," in *Proc. of ACM MobiCom*, vol. 3, no. 4, 2017.
[59] T. Wei and X. Zhang, "Pose Information Assisted 60 GHz Networks: Towards Seamless Coverage and Mobility Support," in *Proc. of ACM MobiCom*, 2017.
[60] S. Sur, I. Pefkianakis, X. Zhang, and K.-H. Kim, "Towards Scalable and Ubiquitous Millimeter-Wave Wireless Networks," in *Proc. of ACM MobiCom*, 2018.
[61] S. Jog, J. Wang, J. Guan, T. M. H. Hassanieh, and R. R. Choudhury, "Many-to-Many Beam Alignment in Millimeter Wave Networks," in *Proc. of USENIX NSDI*, 2019.
[62] M. Park, P. Gopalakrishnan, and R. Roberts, "Interference Mitigation Techniques in 60 GHz Wireless Networks," *IEEE Communications Magazine*, vol. 47, no. 12, 2009.
[63] C.-S. Sum, X. An, Z. Lan, T. Baykas, J. Wang, R. Funada, M. A. Rahman, H. Harada, and S. Kato, "A Synchronization-Frame-Aided Interference Mitigation Mechanism for Millimeter-wave WPAN," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, 2009.
[69] Y. L. Yong Niu, D. Jin, L. Su, and A. V. Vasilakos, "A Survey of Millimeter Wave Communications (mmWave) for 5G: Opportunities and Challenges," *Wireless Networks*, vol. 21, 2015.
[64] Swetank Kumar Saha and Shivang Aggarwal and Rohan Pathak and Dimitrios Koutsonikolas and Joerg Widmer, "MuSher: An Agile Multipath-TCP Scheduler for Dual-Band 802.11ad/ac Wireless LANs," in *Proc. of ACM MobiCom*, 2019.
[65] Facebook Inc., "Terragraph: Solving the Urban Bandwidth Challenge," 2018. [Online]. Available: https://terragraph.com/
[66] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wanga, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter Wave Mobile Communications for 5G Cellular: It Will Work!" in *IEEE Access*, vol. 1, 2013.
[67] M. Elkashlan, T. Q. Duong, and H.-H. Chen, "Millimeter Wave Communications for 5G: fundmentals: Part 1," in *IEEE Communications Magazine*, vol. 52, no. 9, 2014.
[68] ——, "Millimeter Wave Communications for 5G: fundmentals: Part 2," in *IEEE Communications Magazine*, vol. 53, no. 1, 2015.
[70] X. Wang, L. Kong, F. Kong, F. Qiu, M. Xia, S. Arnon, and G. Chen, "Millimeter Wave Communication: A Comprehensive Survey," in *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, 2018.
[71] T. S. Rappaport, Y. Xing, G. R. MacCartney, A. F. Molisch, E. Mellios, and J. Zhang, "Overview of Millimeter Wave Communications for Fifth-Generation (5G) Wireless NetworksWith a Focus on Propagation Models," in *IEEE Transactions on Antennas and Propagation*, vol. 65, no. 12, 2017.
[72] X. Zhou, J. Yang, M. Chrobak, and Y. Zhang, "Performance-Aware Thermal Management via Task Scheduling," *ACM Transactions on Architecture and Code Optimization*, vol. 7, no. 1, 2010.
[73] M. Chrobak, C. Drr, M. Hurand, and J. Robert, "Algorithms for Temperature-Aware Task Scheduling in Microprocessor Systems," *Springer Publications*, vol. 5034, 2008.
[74] J. Liu and L. Zhong, "Micro Power Management of Active 802.11 Interfaces," in *Proc. of ACM MobiSys*, 2008.
[75] G. Anastasi and M. Conti and E. Gregori and A. Passarella, "802.11 Power-Saving Mode for Mobile Computing in Wi-Fi Hotspots: Limitations, Enhancements and Open Issues," *Springer Link Wireless Networks*, vol. 14, no. 6, 2008.
[76] E. Rozner, V. Navda, R. Ramjee, and S. Rayanchu, "NAPman: Network-Assisted Power Management for WiFi Devices," in *Proc. of ACM MobiSys*, 2010.
[77] K. Flautner, S. Reinhardt, and T. Mudge, "Automatic Performance Setting for Dynamic Voltage Scaling," in *Proc. of ACM MobiCom*, 2001.
[78] Xinyu Zhang and Kang G. Shin, "E-MiLi: Energy-Minimizing Idle Listening in Wireless Networks," in *Proc. of ACM MobiCom*, 2011.
[79] B. Gao, Z. Xiao, L. Su, D. Jin, and L. Zeng, "Energy-Efficient Idle Listening Scheme using 1 Bit Sampling in 60 GHz Wireless Local Area Network," *IET Communications*, vol. 9, pp. 219–226, 2015.
[80] Qualcomm Incorporated, "QCA9500," 2018. [Online]. Available: https://www.qualcomm.com/products/qca9500
[81] Daniel Halperin and Wenjun Hu and Anmol Sheth and David Wetherall, "Predictable 802.11 Packet Delivery fromWireless Channel Measurements," in *ACM SIGCOMM*, 2010.
[82] C. Kim, T. Kim, and J.-Y. Seol, "Multi-Beam Transmission Diversity with Hybrid Beamforming for MIMO-OFDM Systems," in *IEEE Globecom Workshops, Atlanta, GA*, 2013.
[83] ASUSTeK Computer Inc., "ASUS ROG: Command and Control: how to set up your own gaming server," https://rog.asus.com/articles/gaming/command-and-control-how-to-set-up-your-own-gaming-server/, 2016.
[84] Google, "Google Cardboard," https://arvr.google.com/cardboard/, 2020.