d-Cellular: Trust-Free Connectivity in Decentralized Cellular Networks

Serhat Arslan
Dept. of Electrical Engineering
Stanford University
sarslan@stanford.edu

Ali Abedi
Dept. of Electrical Engineering
Stanford University
abedi@stanford.edu

Sachin Katti
Dept. of Computer Science
Stanford University
skatti@stanford.edu

***Abstract*—Decentralized cellular networks distribute infrastructure ownership among small, independent entities and lower the barrier to entry into the telecommunication landscape. Unlike traditional subscription models, users of these networks spontaneously connect to any provider in their vicinity without a legal agreement. However, the absence of established trust between users and providers is concerning for service assurance and appropriate payments. Two-sided measurements, where users independently measure the received service to ensure its fulfillment and providers measure it for accurate billing, can address this concern. With this approach, users can also request tailored Service Level Agreements (SLAs) optimized for the performance needs of their applications while verifying them. We design the d-Cellular framework that employs innovative two-sided measurement techniques to verify such complex SLAs. It maps the noisy physical layer telemetry from base stations to the high-level performance measurements of user applications in order to obtain consistent two-sided measurements between users and providers. Our over-the-air evaluations with commercial smartphones and dynamic channel conditions confirm that it can simultaneously deliver complex SLAs to multiple users.**

***Index Terms*—Next-G Architecture, On-Demand Slicing**

I. Introduction

About 5 billion people use the Internet regularly, making it ubiquitous and essential to everyday life, commerce, and education. Yet, most countries are serviced by only a handful of Mobile Network Operators (MNOs) providing users with little or no choice of provider, and little incentive for mobile service providers to invest in new infrastructure or services [1]. This leads to widespread complaints about the quality of voice calls, messaging, and data [2].

Many governments are therefore interested in creating more consumer choice by making it easier for new MNOs to enter the market. However, deploying a country-wide network is expensive, particularly when needing to compete against existing, profitable MNOs. What if we could enable new operators to enter the market without needing to be countrywide? If this were possible, we could lower the barrier to entry for new entrants, allowing them to start - and perhaps stay - small, without the huge investment needed for nationwide scale. This is the idea behind so-called *decentralized wireless* (DeWi), which aims to allow individuals and small businesses to become service providers by deploying just a handful of base stations (BS) [3].

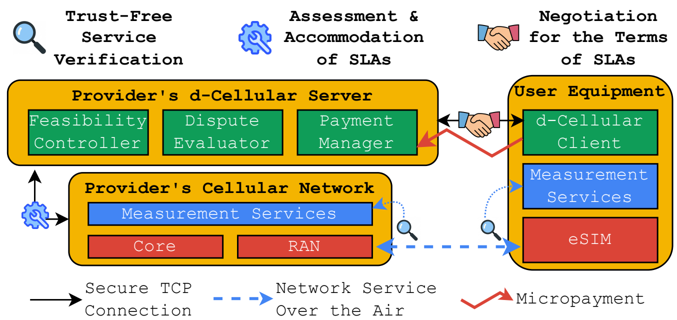

Figure 1: Overview of how d-Cellular addresses the challenges to offering trust-free on-demand slicing.

In such a DeWi network, a user joins whichever local infrastructure is available, based perhaps on the offered data rate, latency, or price. The key to DeWi is that the user dynamically chooses which network to join, creating competition among operators, large and small.

For this to work, the user and the DeWi operator need to authenticate each other, the operator needs to offer a service to the user, and the user needs to decide if the operator is providing the service as promised while appropriately paying for the service provided. All of this requires an element of trust between users and providers, as well as a means to pay for service, leading to the main question we address: *Without any prior legal contract between users and providers, how can users know they are receiving the promised service, and how can providers know they will be paid for the service they provide?*

We posit that a commercial DeWi user needs a lightweight means to verify, in real-time, that the provider delivers the promised service. To do this, we propose a novel *two-sided measurement* technique. Users and providers independently measure the performance and verify the service in small units without the intervention of any third party. The user continues to use the provider if, and only if, their measurements confirm the service is correctly provided. Similarly, the provider only continues to provide service if the user continues to pay. Hence, they are incentivized to be honest, or else the other party will discontinue service.

Essentially, providers need to offer simple, on-demand SLAs to users, and both parties need a means to verify whether or not the SLA was met. For example, self-driving vehicles benefit from low-latency services for cloud-based AI inference. However, during software upgrades, they would temporarily switch providers or the SLA with the same provider for verifiable high-throughput service. This faces some challenges. First, User Equipment (UE) typically does not expose low-level wireless channel telemetry. Second, the telemetry data gathered by providers is typically quite noisy. Therefore, it is hard to compare measurements from both sides; we need to reconcile the differences. Third, UEs lack a means to dynamically declare the SLA they need. Finally, a service provider needs a means to judge whether or not an SLA is feasible to be offered, which is especially difficult during dynamic wireless channel conditions.

We set out to design a system that will allow users and operators to negotiate an SLA and then verify whether or not it was provided, without needing to establish a contract in advance. We call our system d-Cellular, and it is shown in Figure 1. Users and providers first negotiate trust-free smart contracts for custom SLAs () and d-Cellular servers map high-level user requests to low-level resource allocation to determine whether or not the SLA can be met (). If feasible, the provider accepts the SLA and schedules wireless resources accordingly. Then, during operation, the user and provider each run two-sided measurements to verify the performance without requiring a trusted third party (). If users are not satisfied with their measurements, they dispute or simply stop using the service. Providers evaluate the disputes based on their side of the measurements and decide whether to refund or cancel the SLA.

We implemented d-Cellular on commercial smartphones without any hardware modifications or root access while being compatible with all mobile operating systems. We also built a cellular BS using a software-defined radio and an open-source radio suite. Our over-the-air experiments with highly variable wireless channel conditions confirm that despite these challenging conditions, d-Cellular can deliver latency and bitrate-based SLAs to multiple UEs with high confidence.

II. Decentralization with d-Cellular

d-Cellular leaves the UE networking stack and the data plane of service providers as specified in the 3GPP standards for backward compatibility while using open interfaces for service measurements and control [4]. It leverages a network function to let users continuously interact with the control plane of the network and dynamically deploy the QoS-optimized custom slices without legal subscriptions.

Before any SLA is served, UEs are authenticated into the network and provisioned only for sending traffic to d-Cellular servers. d-Cellular clients of UEs use this provision to negotiate the desired performance with the servers. d-Cellular servers map the SLA requests to the physical resources needed at the associated BS in order to decide whether they are feasible. If feasible, d-Cellular periodically configures the BS to ensure the network performance is within the negotiated budgets despite the dynamic wireless channel conditions.

Both the BS and the UE use two-sided measurement techniques described in §II-A to monitor the performance over time. Hence, they can verify the service and make sure it meets the negotiated SLA without any trust. As the service is verified, the UE periodically pays the provider to continue the service. Alternatively, it disputes the SLA and does not pay for the next period or switch providers for a better service. If the provider's measurements agree with the dispute, the next period is served for free to prevent customer churn. Otherwise, the SLA is canceled due to the missing payment after a timeout.

Each payment transaction is called a *micropayment* and includes a cryptographic signature to prove that the client accepts to pay. The signatures are accumulated in state channels to settle on a blockchain after a billing period [3]. Alternatively, the provider can charge the users with conventional methods like credit cards in existing MNOs.

Note that a 100% performance guarantee is impossible in a mobile setting due to uncontrollable wireless channel conditions. d-Cellular adopts a middle ground for this challenge. Declaring user requests is an opportunity for providers to attempt satisfying them without a guarantee which is incentivized by incremental payments that are fairly adjusted after two-sided verification. Plus, the duration of each SLA is short enough to limit UE's loss in dollar value when a provider fails to satisfy it.

A. Metrics for Two-Sided Measurements & SLA Verification

Next, we list the 2 main types of SLAs and describe how d-Cellular servers measure them.

1) Latency SLAs: d-Cellular is the first proposal to verify end-to-end latency for cellular network users. ITU G.114 recommends mouth-to-ear one-way delay for real-time video conferencing to be less than 400 msec, i.e., 800 msec of Round Trip Time (RTT). Discounting the processing delays at the end hosts and the application servers leaves a network RTT allowance of 200 to 300 msec depending on the video conferencing application [5]. Thus, users would be interested in latency SLAs that can help achieve RTTs lower than 300 msec. d-Cellular addresses this challenge by monitoring the fraction of the RTT corresponding to the provider's network and controlling this bit to satisfy the end-to-end budget.

In return, users declare their intended bitrate for the SLA to guarantee that they will not over-utilize all the buffer space allocated to them. This is consistent with the fact that low latency application traffic is mostly application limited [6]. Otherwise, buffer-filling congestion control algorithms like CUBIC increase RTT as much as the bottleneck buffer size which inherently maximizes the latency at all times.

Figure 2 summarizes the overall procedure for controlling the latency of the SLA traffic. d-Cellular's novel latency verification technique breaks down RTT into 3 elements with the following relationship:

$$RTT_{SLA} = RTT_{WAN} + RTT_{MNO} + RTT_{AIR} \quad (1)$$

RTT_{WAN} is the latency between the edge of the provider's network and the destination of the traffic whereas RTT_{MNO}

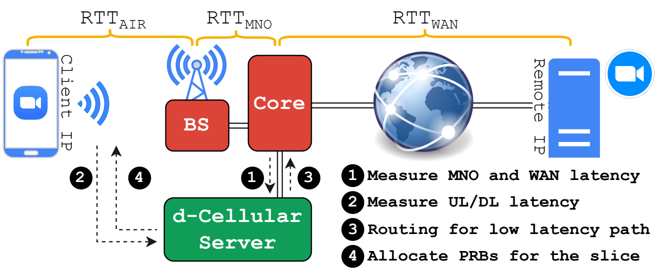

Figure 2: Fractions of RTT for d-Cellular's latency SLAs.

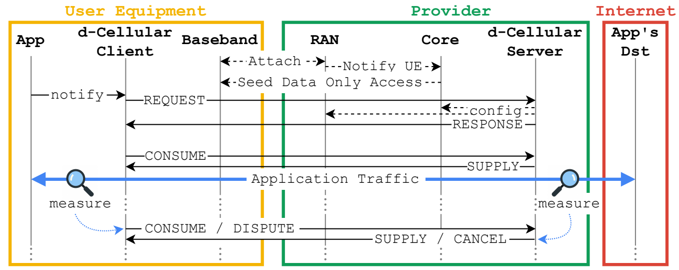

Figure 3: Communication pattern for d-Cellular negotiations.

is the latency between the core and the BS. Once an SLA is requested, d-Cellular servers start periodically measuring these values by sending out ping packets from the provider's edge to the *remote IP* address and the BS, respectively. Providers also minimize them by routing traffic to the path with the lowest latency. The period of probes is set to 500 msec to avoid an extensive amount of probe packets at a time.

Once RTT_{WAN} and RTT_{MNO} are known, the budget for the RTT_{AIR} is calculated with (1). It is the latency between the UE and the BS over the wireless channel and can be controlled by changing the Physical Resource Block (PRB) allocation over the spectrum for the UE when the wireless link is the bottleneck for the traffic.[1] Larger number of PRBs increases the bandwidth of the channel for the UE, leading to less queuing and smaller RTT_{AIR}. When the average RTT_{AIR} is below the budget, the server de-allocates some PRBs to utilize them for other users. These measurements and reallocations are frequently repeated to keep the latency under the budget with minimal use of PRBs.

2) Usage & Bitrate SLAs: The most popular metric for billing in conventional networks is the usage, i.e., the amount of data downloaded or uploaded. d-Cellular servers periodically query the current bitrate from the BS. As the bitrate for each UE is measured, a server dynamically increases or decreases PRB allocations for each user's slice to make sure the performance budgets are satisfied. The lower-bound on a budget indicates the minimum bitrate requested by the user for an acceptable experience. The upper-bound of a budget denotes the bitrate after which the user would gain negligible marginal benefit. It mainly helps the providers to figure out minimal resource allocation while optimizing user experience.

B. Network Function to Assess and Accommodate SLAs

Once a request for an SLA is received by the servers, determining the feasibility of the requested SLA is not a straightforward procedure because there is no guarantee that the current channel conditions will persist throughout the SLA period. 3GPP Spec 36.213 provides tables to map the current Modulation Coding Scheme (MCS) for UEs and the number of allocated PRBs to the expected bitrate for the connection. Servers use the tables in reverse to calculate the minimum number of PRBs required for the negotiated bitrate given the MCS even for latency SLAs. This gives the *principal number* of the SLA and d-Cellular oscillates the actual PRB allocations around this number to control the primary metric of the SLA. An SLA is declared infeasible if its principal number plus the principal numbers of the other active SLAs exceed the bandwidth capacity of the BS.

The server queries BS for telemetry every 500 msec to reallocate resources and satisfy SLAs in case the channel conditions have changed. The time between each query is the *reallocation period*. A shorter period enables more accurate decisions for the PRB allocations. However, it trades off higher computation and signalling overhead for the reallocation.

If the number of PRBs calculated for all UEs exceeds the bandwidth capacity of the BS during an SLA period, some UEs end up getting fewer PRBs than required to satisfy their SLA. Then, the server cancels failing SLAs at the end of the SLA period which is chosen short to minimize the dollar value of each failed period. We heuristically choose 30 seconds as the SLA period in our implementation.

Note that handover or roaming is a frequent procedure in mobile networks. Then, an automated collaboration between independent providers, i.e., roaming without prior agreements, is needed. d-Cellular servers may choose to communicate with other servers that manage the neighboring providers to reserve PRBs in advance and forward packets of the users during handovers. d-Cellular design allows negotiating these handover reservations just like SLA negotiations. We leave this to providers as a business decision for future work.

C. Negotiating for the SLAs

The negotiations between the servers and the clients are conducted with d-Cellular messages. The negotiation pattern is shown in Figure 3 with the message contents in Listing 1.

Listing 1: d-Cellular message content

```
'msg_type'    : The purpose of the message
'client_ip'   : Uniquely identifies the UE
'sla_id'      : Uniquely identifies the SLA
'sla_period'  : Duration of each SLA
'sla_type'    : Primary metric of the SLA
'remote_ip'   : Remote end of the SLA traffic
'budgets'     : Requested performance ranges
'price'       : Cost of the SLA
'timestamp'   : Time of the message
'signature'   : Cryptographic proof of payment
```

There are 6 types of messages in d-Cellular:

1) ***REQUEST****:* Clients send this to initiate a negotiation for the given *SLA type*. The server may consider the *remote IP*

[1] If the wireless link is not the bottleneck and RTT exceeds the budget, the provider declares the SLA as infeasible.

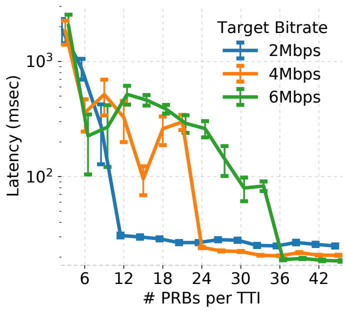

Figure 4: Effect of PRB allocation on the achieved latency.

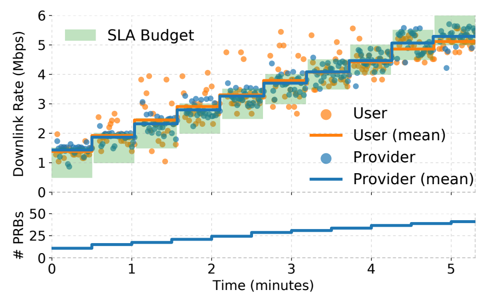

Figure 5: Bitrate SLA budgets and the two-sided measurements.

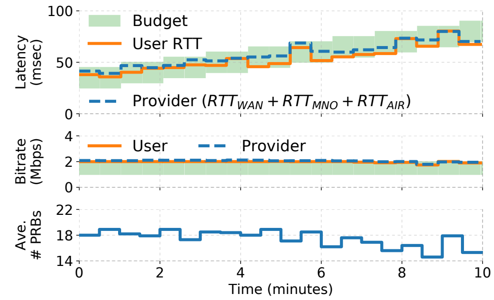

Figure 6: Latency SLA budgets and the two-sided measurements.

address to figure out the end-to-end performance limits for flows destined to this address, i.e., RTT_{SLA}.

2) ***RESPONSE****:* Servers send this when a request is accepted. It confirms the feasible performance *budgets* while declaring the *price* and the *SLA period*. A unique *SLA ID* is shared for the client to use in future payments for this SLA.

3) ***CONSUME****:* Asks providers to start serving the given *SLA ID*. It is re-sent by the client every *SLA period* as long as the client is satisfied and wants to continue. The *signature* of the client cryptographically confirms the accepted *price* to be paid while the *timestamp* prevents replay attacks by adversaries.

4) ***DISPUTE****:* Expresses the client's dissatisfaction with measurements in the previous *SLA period* with either partial or no payment for the next period. The server may either accept the partial payment and supply the service or cancel the SLA.

5) ***SUPPLY****:* Sent by servers to mark the beginning of the *SLA period* per the received CONSUME. The *price* field indicates the received payment for the current period in case it was partial or none due to a dispute. The service is supplied for the duration of *SLA period* plus a grace period to avoid any re-negotiating if the next CONSUME is delayed, enabling uninterrupted service to UEs over multiple SLA periods.

6) ***CANCEL****:* Servers cancel SLAs when a UE disputes but the provider measures satisfactory performance. It is also sent after a timeout with no payments from the client.

III. Prototyping and Evaluation

We implement d-Cellular for both emulations and over-the-air experiments. The over-the-air setup utilizes a BladeRF xA9 software-defined radio as the BS, and Samsung Galaxy S20 and S8 as UEs to evaluate d-Cellular's compatibility with non-emulated channel conditions and equipment.[2] The software-defined radio nodes in both of the setups use srsRAN's open-source networking stack [7]. The low-level BS telemetry for d-Cellular is extracted using the SCOPE framework [8].

We evaluate d-Cellular for (i) its capability to determine feasible SLA budgets (§III-A), (ii) its effectiveness to derive agreeable performance metrics out of imperfect two-sided measurements (§III-B), (iii) its overall performance with dynamic channel conditions (§III-C).

[2]The artifact is published at https://github.com/serhatarslan-hub/d-cellular

A. Determining the Feasibility of SLAs

We evaluate how precisely the calculated principal number reflects the actual bitrate measured by the user while emulating a real-world scenario. The UE is stationary and the channel has an MCS value of 11 in this scenario. We run TCP iPerf traffic between the provider's core and the user through the downlink while incrementing the number of allocated PRBs every 30 seconds. As the number of PRBs increases, the bitrate also increases for a given MCS. Then, we measure the achieved throughput at the UE every second and compare it against the expected rate. Our data show that UE's measurements are steadily 4.5% lower on average than the expected rate due to the overhead of packet headers and control traffic. This means the bitrate SLAs can be declared feasible if the provider has principal number of available PRBs that are not reserved for other SLAs plus an extra as a safety margin.

Next, we investigate how allocated PRBs affect latency by repeating the previous scenario with different target bitrates for a TCP flow. The target bitrate limits the throughput of the flow, but the achieved throughput will be lower if the network capacity is lower. We choose 2, 4, and 6 Mbps as targets to represent typical rates for low latency real-time video conferencing applications [6]. The theoretical principal numbers in this experiment are 12, 23, and 35 respectively.

We explore the relationship between the principal numbers and the latency by allocating one more PRB every 30 seconds while measuring RTT at the UE. Figure 4 shows the average RTT for different numbers of PRBs with error bars indicating the standard deviation of the measurements. When less than the principal number of PRBs are allocated, the wireless link becomes the bottleneck and the BS queues up packets causing high variance in the latency. On the other hand, the BS can sustain the target rate without queuing when more than the principal number of PRBs are allocated. We conclude that the desired latency can be achieved by controlling the allocated number of PRBs around this principal number. Moreover, the latency SLAs can be declared feasible if the BS is available to allocate at least *"principal number + 1"* PRBs to the UE.

B. Composing Mutually Acceptable Two-Sided Measurements

We repeat the scenario in §III-A with successive SLA negotiations to evaluate how two-sided measurements match.

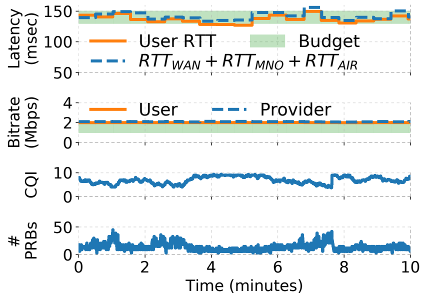

Figure 7: Performance with latency SLA for a moving UE.

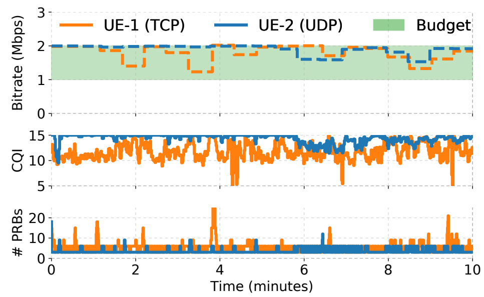

Figure 8: Performance for bitrate SLAs with 2 UEs over-the-air.

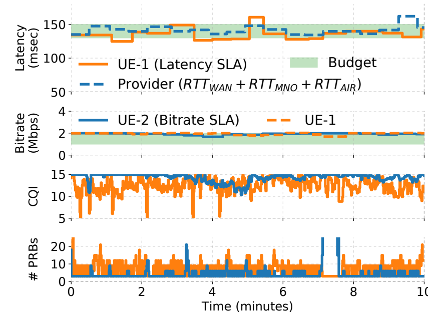

Figure 9: Performance for latency and bitrate SLAs with 2 UEs over-the-air.

The user artificially negotiates for a higher bitrate in each SLA period to test if the d-Cellular server can adapt to these changing SLAs. Figure 5 shows the two-sided measurements for this scenario. Due to fluctuations in the wireless link, bitrate measurements at every reallocation period (denoted as points on the figure) are noisy and sometimes out of budget. In addition, individual measurements from the BS and the UE may not precisely match due to slightly different measurement times or errors. To overcome these challenges, d-Cellular considers the average performance over an SLA period. Hence, only the mean of the measurements is plotted for each period in the rest of the evaluations. The orange and blue lines in Figure 5 show that the average bitrate at the UE and BS are always within 5% of each other and within the budgets for all the SLAs. The figure also shows that as the bitrate requested by the UE goes up, the server correctly increases the number of PRBs allocated to this UE to match the target bitrate.

Next, we evaluate the efficacy of d-Cellular's novel two-sided measurements for latency SLAs with a scenario similar to the one described above. In this experiment, the bitrate target is fixed to the range of 1 to 2 Mbps and the latency requirement is changed in every SLA period. Figure 6 shows how well the two-sided measurements match and the average number of PRBs allocated in each SLA period. The orange line on top represents the UE's RTT measurements whereas the dashed blue line is the provider-side measurements per (1) and the middle plot represents the measured bitrate. As the latency requirement is relaxed, the d-Cellular server decreases the average number of PRBs for this UE to increase the latency and keep it within the requested budget.

C. Overall performance of d-Cellular

To evaluate d-Cellular in realistic scenarios, we investigate what kind of latency budgets would be useful to users while being feasible for providers by analyzing Zoom traffic on our campus as a good use case for latency SLAs. We measure the RTT_{WAN} for the application traffic and calculate the remaining budgets for RTT_{AIR} as described in §II-A1. Since the user-to-user network RTT allowance is typically 300 msec (see §II-A1), clients can have a round-trip latency of 150 msec to a Zoom server, which is also endorsed by the company as well [6]. However, RTT_{WAN} for 90^{th}-p of our campus users is around 120 msec in our traces. RTT_{AIR} budget for these users would be in a range less than 30 msec which is shorter than what 80% of LTE users experience in general [9]. Hence, they would benefit from latency SLAs with an upper bound of 150 msec RTT latency. For the rest of our evaluations, we artificially set the RTT_{WAN} for our setups as 120 msec using Linux's traffic controller `tc` and negotiate SLAs for 150 msec of RTT to mimic those users.

In our experiments, UEs negotiate a single SLA for each experiment which is sustained with `CONSUME`/`SUPPLY` messages without any service disruptions. Figure 7 shows the performance of d-Cellular in emulations where we measure the delay and throughput performance obtained by a single UE with a latency SLA moving at the walking speed for 10 minutes. The two-sided measurements of the user and provider match and the performance metrics are always within the negotiated budget throughout the experiments. As depicted by the significant Channel Quality Index (CQI) changes, wireless channel conditions are highly variable in these scenarios.

Next, we evaluate d-Cellular with over-the-air scenarios. We set up a BS using a software-defined radio along with commercial smartphones. In these experiments, multiple phones are connected to the BS and carried around for 10 minutes. The users' movement consists of a combination of walking and standing. Figure 8 shows the performance of d-Cellular when multiple UEs request bitrate SLAs for UDP and TCP-based traffic with a budget of 1 to 2 Mbps. This budget reflects typical bitrates required by real-time video conferencing applications [6]. The figure shows that the CQI fluctuates significantly, especially for UE-1. Despite these variable channel conditions, d-Cellular can successfully change the number of assigned PRBs in a way that the performance is kept within the negotiated budget for both UEs at all times.

In the final experiment, UE-1 requests a TCP-based latency SLA and UE-2 requests a bitrate SLA for UDP traffic with highly variable channel conditions. The budgets are 130-150 msec and 1-2 Mbps for latency and bitrate, respectively. Figure 9 shows that the achieved latency for UE-1 is mostly within the budget. In a few cases, e.g., around minute 1, d-Cellular over achieves the SLA where the latency is better (i.e., lower) than the negotiated budget. This is typically fine for users and d-Cellular clients do not dispute it. On the other hand, d-Cellular underperforms around minute 5 and the latency goes slightly above the budget. This is mainly due to the rare glitches in two-sided measurements, i.e., kernel delays for UE's measurements which are not reflected in the provider's. In the meantime, UE-2 (i.e., bitrate SLA) performs within the budget for the entire experiment.

IV. Limitations of d-Cellular

d-Cellular is the first proposal for verifying end-to-end latency in cellular networks. However, an even more diverse set of SLAs can be defined if new ways of measuring performance both at the UE and the BS are standardized. Otherwise, d-Cellular assumes that UEs can only negotiate for SLAs that can be verified with their end-to-end measurements. Then, providers are expected to map end-to-end performance to their wireless resource allocations when determining the feasibility of the requests and verifying the SLAs.

The dynamic nature of wireless links makes it hard to achieve deterministic behavior for end-to-end performance. Therefore, d-Cellular may not be able to guarantee a 100% satisfaction rate of SLAs in some edge cases. If the BS does not have more PRBs to allocate for the UE when the channel conditions degrade, the SLA may not be satisfied. We assert that changes in the channel conditions are neither the provider's nor the UE's fault. Therefore, both parties should accept that the SLA is not feasible anymore and stop negotiating for some number of periods in the future.

V. Related Work

Decentralized cellular networks are attracting the attention of researchers and entrepreneurs [10]–[13]. Current proposals require third-party legal trust brokers [14], [15]. d-Cellular does not require trusted third parties and enables tailored, on-demand services beyond best effort.

[3] and [15] leverage smart contracts on blockchains and well-defined service measurements to detect underperforming SLAs in a deterministic manner. But they only describe a preliminary design without any details about how negotiation occurs, how providers accommodate SLAs, and the types of SLAs that can be offered. To the best of our knowledge, d-Cellular is the first end-to-end, trust-free system design to address all these questions at once.

VI. Conclusion

If we can make it easier for more mobile operators to get off the ground, even at a small scale (e.g., a college campus, a local government, or a railway station), the cellular service would become more competitive, higher quality, and advance more rapidly. We are far from this situation today. If users and cellular service providers can have short-lived contracts, without any prior relationship, and if both parties are incentivized to be honest; then it may encourage small providers to enter the market, providing the bare minimum infrastructure. If they offer better service than the large incumbents, they can create loyalty among their users. If they can be nimble, for example by providing innovative new services, then they can even provide competitive pressure for all MNOs to improve service over time. This, we believe, is the missing piece in today's cellular networks.

References

[1] H. Trostle, C. Mitchell, N. O. Razafindrabe, M. Andrews, and K. Kienbaum, "Profiles of Monopoly: Big Cable and Telecom," Aug. 2020. [Online]. Available: https://ilsr.org/wp-content/uploads/2020/08/2020_08_Profiles-of-Monopoly.pdf

[2] G. Effler, "Failure to Load: More Device Usage by Wireless Customers Means More Streaming and Data Problems," Jul. 2022, j. D. Power. [Online]. Available: https://www.jdpower.com/business/press-releases/2022-us-wireless-network-quality-performance-study-volume-2

[3] S. Anand, S. Arslan, R. Chopra, S. Katti, M. K. Vaddiraju, R. Rana, P. Sheng, H. Tyagi, and P. Viswanath, "Trust-Free Service Measurement and Payments for Decentralized Cellular Networks," in *Proceedings of the 21st ACM Workshop on Hot Topics in Networks*, ser. HotNets '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 68–75. [Online]. Available: https://doi-org.stanford.idm.oclc.org/10.1145/3563766.3564093

[4] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "Understanding O-RAN: architecture, interfaces, algorithms, security, and research challenges," *CoRR*, vol. abs/2202.01032, 2022. [Online]. Available: https://arxiv.org/abs/2202.01032

[5] A. Jones, P. Sevcik, and R. Wetzel, "Internet Connection Requirements for Effective Video Conferencing to Support Work from Home and eLearning," NetForecast Inc., Technical Report, Feb. 2021. [Online]. Available: https://www.netforecast.com/wp-content/uploads/NFR5137-Videoconferencing_Internet_Requirements.pdf

[6] Zoom Video Communications Inc., "Accessing meeting and phone statistics," Sep. 2022. [Online]. Available: https://support.zoom.us/hc/en-us/articles/202920719-Accessing-meeting-and-phone-statistics

[7] Software Radio Systems (SRS), "srsRAN: Your own mobile network," 2014. [Online]. Available: https://www.srslte.com

[8] L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "SCOPE: An Open and Softwarized Prototyping Platform for NextG Systems," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 415–426. [Online]. Available: https://doi.org/10.1145/3458864.3466863

[9] M. Laner, P. Svoboda, P. Romirer-Maierhofer, N. Nikaein, F. Ricciato, and M. Rupp, "A comparison between one-way delays in operating HSPA and LTE networks," in *2012 10th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, 2012, pp. 286–292. [Online]. Available: https://ieeexplore.ieee.org/document/6260469

[10] Y. Li, K.-H. Kim, C. Vlachou, and J. Xie, "Bridging the Data Charging Gap in the Cellular Edge," in *Proceedings of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 15–28. [Online]. Available: https://doi.org/10.1145/3341302.3342074

[11] M. Johnson, S. Sevilla, E. Jang, and K. Heimerl, "DLTE: Building a More WiFi-like Cellular Network: (Instead of the Other Way Around)," in *Proceedings of the 17th ACM Workshop on Hot Topics in Networks*, ser. HotNets '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 8–14. [Online]. Available: https://doi.org/10.1145/3286062.3286064

[12] J. Ernst, Z. D. Wang, S. Abraham, J. Lyotier, C. Jensen, M. Quinn, and D. Harvey, "A Decentralized Mobile Mesh Networking Platform Powered by Blockchain Technology and Tokenization," RightMesh AG, White Paper, Jul. 2019. [Online]. Available: https://rightmesh.io/docs/RightMesh_WP6.pdf

[13] A. Haleem, A. Allen, A. Thompson, M. Nijdam, and R. Garg, "Helium: A Decentralized Wireless Network," Helium Systems, Inc., White Paper, Nov. 2018. [Online]. Available: http://whitepaper.helium.com

[14] Z. Luo, S. Fu, M. Theis, S. Hasan, S. Ratnasamy, and S. Shenker, "Democratizing Cellular Access with CellBricks," in *Proceedings of the ACM SIGCOMM '21 Conference*. New York, NY, USA: Association for Computing Machinery, 2021, p. 626–640. [Online]. Available: https://doi.org/10.1145/3452296.3473336

[15] T. Faisal, D. Di Francesco Maesa, N. Sastry, and S. Mangiante, "AJIT: Accountable Just-in-Time Network Resource Allocation with Smart Contracts," in *Proceedings of the ACM MobiArch 2020 The 15th Workshop on Mobility in the Evolving Internet Architecture*, ser. MobiArch'20. New York, NY, USA: Association for Computing Machinery, 2020, p. 48–53. [Online]. Available: https://doi.org/10.1145/3411043.3412506