

ExPECA: An Experimental Platform for Trustworthy Edge Computing Applications

Samie Mostafavi
KTH Royal Institute of Technology
Stockholm, Sweden
ssmos@kth.se

Vishnu Narayanan Moothedath
KTH Royal Institute of Technology
Stockholm, Sweden
vnmo@kth.se

Stefan Rönngren
KTH Royal Institute of Technology
Stockholm, Sweden
steron@kth.se

Neelabhro Roy
KTH Royal Institute of Technology
Stockholm, Sweden
nroy@kth.se

Gourav Prateek Sharma
KTH Royal Institute of Technology
Stockholm, Sweden
gpsharma@kth.se

Sangwon Seo
KTH Royal Institute of Technology
Stockholm, Sweden
sangwon@kth.se

Manuel Olguín Muñoz
KTH Royal Institute of Technology
Stockholm, Sweden
manuel@olguinmunoz.xyz

James Gross
KTH Royal Institute of Technology
Stockholm, Sweden
jamesgr@kth.se

ABSTRACT

This paper presents ExPECA, an edge computing and wireless communication research testbed designed to tackle two pressing challenges: comprehensive end-to-end experimentation and high levels of experimental reproducibility. Leveraging OpenStack-based Chameleon Infrastructure (CHI) framework for its proven flexibility and ease of operation, ExPECA is located in a unique, isolated underground facility, providing a highly controlled setting for wireless experiments. The testbed is engineered to facilitate integrated studies of both communication and computation, offering a diverse array of Software-Defined Radios (SDR) and Commercial Off-The-Shelf (COTS) wireless and wired links, as well as containerized computational environments. We exemplify the experimental possibilities of the testbed using OpenRTiST, a latency-sensitive, bandwidth-intensive application, and analyze its performance. Lastly, we highlight an array of research domains and experimental setups that stand to gain from ExPECA's features, including closed-loop applications and time-sensitive networking.

CCS CONCEPTS

• **Networks** → **Network experimentation; Wireless access points, base stations and infrastructure;** • **Computer systems organization** → *Distributed architectures.*

KEYWORDS

Edge computing experimental platform, reproducibility, end-to-end experimentation, wireless testbed

ACM Reference Format:

Samie Mostafavi, Vishnu Narayanan Moothedath, Stefan Rönngren, Neelabhro Roy, Gourav Prateek Sharma, Sangwon Seo, Manuel Olguín Muñoz, and James Gross. 2023. ExPECA: An Experimental Platform for Trustworthy Edge Computing Applications. In *The Eighth ACM/IEEE Symposium on Edge Computing (SEC '23)*, December 6–9, 2023, Wilmington, DE, USA. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3583740.3626819>

1 INTRODUCTION

As we move onto the next decade, edge computing applications are poised for significant growth, fueled by advancements in wireless technologies such as 5G and Beyond 5G (B5G), as well as breakthroughs in compute-intensive tasks like model predictive control, multimedia processing, and machine learning algorithms. In order to achieve trustworthiness and dependability in these interdisciplinary, distributed systems, involving computer science and wireless research, it is imperative to develop sophisticated scientific tools. On one hand, research in areas like wireless networking necessitates close interaction amongst radio hardware with a high degree of adaptability, as facilitated by Software Defined Radios (SDRs). Conversely, the computational aspect requires performance workloads to be adaptable to fluctuating network conditions, necessitating the creation and efficient repeatable testing of innovative computing algorithms. End-to-end experimentation is essential for obtaining a comprehensive understanding of the system as a whole and for devising innovative approaches to overcome bottlenecks. Moreover, the ability to efficiently replicate these experiments is key to allowing other researchers to validate the findings.

In this realm of research, we find a diverse landscape of testbeds that have emerged in recent years, each distinguished by its size, applicability, and feature set. Notable ones include the (1) COSMOS, (2) POWDER, (3) ARA, and (4) Drexel Grid SDR testbeds. COSMOS (Cloud enhanced Open Software defined MOBILE wireless testbed for city-Scale deployment) is a testbed spanning over 1 square mile, equipped with an array of resources, including Software-Defined Radios (SDRs), millimeter-wave (mmWave) equipment, optical fibers,

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
SEC '23, December 6–9, 2023, Wilmington, DE, USA
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0123-8/23/12.
<https://doi.org/10.1145/3583740.3626819>

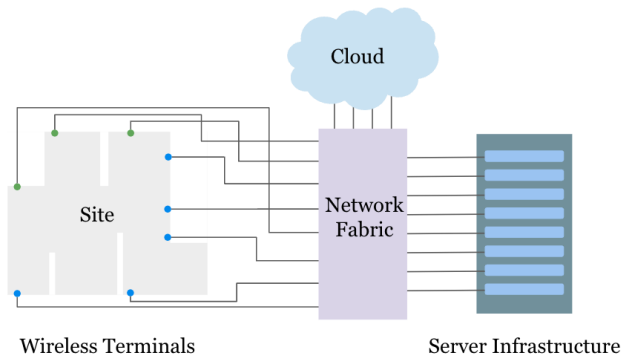


Figure 1: Conceptual architecture of ExPECA for end-to-end edge computing experiments.

and various computational nodes for core functionality and application processing [17, 20]. COSMOS integrates with cloud infrastructure, with a primary focus on advancing cloud computing research. This testbed includes a diverse set of rooftop, intermediate, and mobile nodes, all centrally controlled.

POWDER (Platform for Open Wireless Data-driven Experimental Research) serves as an even larger testbed, offering a platform for exploring wireless and mobile networks with a high level of programmability that extends down to the waveform level [2]. Covering an area of 15 km², POWDER comprises fixed, programmable radio nodes constructed using off-the-shelf SDR technology. The platform seamlessly integrates with cloud and edge-like compute capabilities, allowing mobile nodes to tap into the compute resources.

The Drexel Grid SDR testbed features SDRs that connect over-the-air (OTA), channel emulator, or alternatively a combination of the two, to enable realistic and reproducible experimentation. Even though the testbed provides freedom in terms of experiment development, it is intended primarily for SDR-related research, and does not integrate any core, cloud or edge components.

ARA (The Agricultural and ruRAL communities) represents a wireless research platform, spanning a rural expanse with a diameter of 60 km in Iowa [21]. ARA’s mission is to investigate wireless platforms and technologies within the real-world context of agricultural settings. The testbed incorporates a diverse array of wireless platforms, implemented through SDRs and programmable Commercial Off-The-Shelf (COTS) radios along with automated vehicles and various cameras and other sensors. ARA’s testbed software builds upon the Chameleon Infrastructure (CHI) software framework from the Chameleon testbed project [7]. Chameleon primarily serves as a bare-metal reconfigurable research testbed designed for edge-to-cloud experimentation, adapting the cloud paradigm for computer science research. Since its public unveiling in 2015, Chameleon has accommodated over 6,000 users across more than 800 projects, gaining valuable experience and refining methods for automating and managing research infrastructure sites. This accumulated knowledge has been encapsulated in the CHI-in-a-Box [6] testbed software distribution, which aims to simplify and optimize the deployment and operation of future associated or independent Chameleon sites.



Figure 2: A photograph of the underground site (at KTH R1) of the testbed.

Existing wireless research-focused platforms like COSMOS, POWDER, and Drexel Grid employ their own specialized software solutions, which are not in line with prevailing cloud-native standards. These testbeds also utilize virtualization technology based on Virtual Machines (VMs), rather than opting for lighter and more edge-compatible solutions such as containers. Similar to the approach taken by the ARA testbed, we found that CHI-in-a-Box presents an excellent opportunity to develop experimental platforms at minimal cost, while offering significant flexibility in both cloud and edge-native technologies. However, when it comes to reproducibility in wireless edge-computing research, existing large-scale testbeds have limitations that we aim to address by introducing our testbed, ExPECA. By ‘reproducibility’, we mean the capacity for researchers to easily replicate experiments and validate results. Achieving this is particularly challenging for two main reasons: the first being, edge computing experiments involve a range of heterogeneous components like servers and radios, all interacting within a single experiment; and second, the existence of wireless links introduces variables such as interference and dynamic propagation conditions that must be controlled for consistent results. Owing to its unique design choices, ExPECA provides a highly reproducible environment for end-to-end wireless and edge computing experimentation, making it one of the first platforms to effectively address this issue. In the following section, we delve into these design intricacies and their consequential impacts.

2 TESTBED DESIGN AND ARCHITECTURE

The ExPECA testbed is designed to address two primary challenges pervasive in edge computing research: 1) end-to-end experimentation, 2) ensuring repeatability and reproducibility. To enable end-to-end edge computing experiments that encompass both communication and computation aspects, ExPECA is engineered with the following primary goals:

- (1) Provisioning communication components by utilizing a diverse array of wireless and wired links.
- (2) Executing and overseeing computational elements within containerized environments running on the testbed infrastructure.

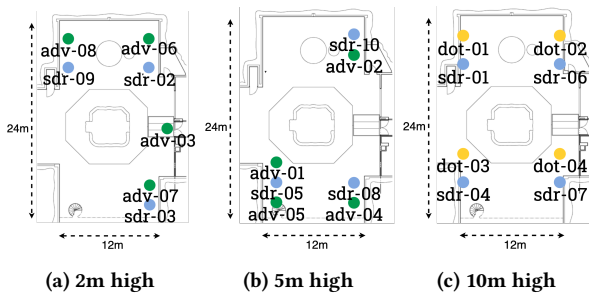


Figure 3: Low, mid, and high-level radio nodes locations at R1 on three different levels as of September 2023.

These objectives are realized through an architectural framework depicted in Figure 1. Central to this architecture is a versatile network fabric that ensures IP connectivity among different experimental components, which may include servers located in the cloud or at the edge, as well as wireless terminals.

Distinctive design choices have been made to realize these objectives, distinguishing ExPECA from contemporary solutions. We here describe these choices and their impacts.

2.1 Location and Physical Environment

The testbed is located in KTH R1 hall, an experimental facility 25 meters below ground level. This isolated location offers a controlled environment with minimal radio interference, making it ideal for running reproducible wireless experiments. With dimensions of 12 meters in width, 24 meters in length, and 12 meters in height, the facility is spacious enough to accommodate a variety of remotely controlled devices, such as drones and cars, providing researchers with considerable freedom in designing their experiments.

2.2 Control and Management Software

We adopted CHI-in-a-Box as a baseline testbed implementation with extensions and contributions from our team. CHI-in-a-Box is a packaging of the CHameleon Infrastructure (CHI) software framework which is built primarily on top of the mainstream open-source OpenStack platform. CHI has mature support for edge-to-cloud provisioning of compute, storage, and network fabrics, and provides end-to-end experiment control and user services. Specifically, we leverage the CHI@Edge flavor of the framework, a novel version that supports running containerized workloads on compute resources. In the following, we describe the essential elements of CHI. Users have the flexibility to allocate resources either on-the-fly or through advance reservations, facilitated by customized versions of OpenStack’s Blazar and Doni services. These services have been tailored by the ExPECA team to include unique features such as radio and Kubernetes node reservations. The scope of allocatable resources encompasses worker nodes, radios, networks, and IP addresses. Once resources are secured, the integrated Kubernetes orchestration system enables the deployment of containerized workloads via OpenStack Zun. Users have the option to utilize pre-configured Docker images supplied by the ExPECA team (as in the case of SDR RAN) or to upload their own container images. Notably, the ExPECA team has extensively modified Zun

to support functionalities like network interface and block storage attachments to K8S containers, features not originally supported in CHI@Edge. Configuration of complex experiment topologies such as distributed networking experiments is supported through CHI’s programmable interface and our adopted flavor of the python-chi Python library. In addition to these, OpenStack Neutron plays a pivotal role in managing the network architecture. Neutron provides a robust set of networking capabilities, including virtual networking, that allows for the creation of isolated networks, subnets, and routers. We connect all testbed components, including radios, to managed Layer 2 switches controlled by Neutron.

2.3 Radio Nodes

One of the standout features of ExPECA is the inclusion of wireless terminals as reservable resources. These radios are mounted at diverse locations in R1, at three different levels, as shown in Figure 3. They are categorized into two types:

- (1) SDR Nodes: These can serve as the air interface for any wireless protocol implemented for USRP E320 SDRs, including OpenAirInterface (OAI) 5G, OAI LTE, SRS LTE, Mangocomm IEEE802.11b/g, or any GNU radio-based wireless protocol.
- (2) COTS Radios: Includes a private 5G system with radio dots and Advantech routers serving as 4G/5G COTS UE nodes.

All radio nodes are equipped with IP-networked interfaces, specifically SDRs. This allowed us to enroll them as network segments in OpenStack Blazar and Doni with minimal extension efforts. The users can easily integrate them into any workload container or SDR host container, through the testbed software. For example, researchers can seamlessly transition from a high-quality 5G link between locations sdr-01 and sdr-06 to a poor-quality channel between sdr-09 and sdr-07, enabling them to empirically assess the impact on application performance. Alternatively, they can switch to WiFi between sdr-01 and sdr-06 for further experimentation.

2.4 Compute Nodes

ExPECA offers compute resources by allowing users run containerized applications with the most recent version of OpenStack Zun that uses K8S (Kubernetes) as the orchestration layer. K8S is designed to handle a large number of containers across a variety of environments, making it well-suited for complex, large-scale deployments. Also, it offers a rich set of APIs and a pluggable architecture, allowing for greater customization and extensibility. This was particularly useful for our team when developing new desired features in Zun. These new features include 1) network attachment, 2) block storage attachment, 3) resource management i.e. specifying the number of cores and memory size for the container. Users are able to run their containerized applications on ExPECA’s Intel x86 servers, enrolled in the K8S cluster, which we refer to as ‘workers’, not only to deploy and orchestrate application workloads, but also to deploy SDR communication stacks or apply radio node configurations. With the advent of Open Radio Access Network (O-RAN) and SDRs, all components of the wireless network stack can run on general-purpose processors. They can thus be containerized and distributed across multiple hosts as long as the containers are connected to the SDRs. We provide reference container images for SDR 5G, SDR LTE, and WiFi provisioning. Lastly, it is important to

note that all worker nodes achieve sub- μ s synchronization by exchanging PTP messages with a Grandmaster (GM) clock that itself uses GPS as a reference. Tight time synchronization is essential for users who require the same timing reference to timestamp packets across various locations within the testbed.

3 EXPERIMENTAL TESTBED VALIDATION

This section aims to substantiate two key assertions about ExPECA: its efficiency in facilitating end-to-end experiments and its ability to ensure experimental reproducibility¹.

To validate the first claim, we employ OpenRTiST [4], an open-source application that is computationally demanding, bandwidth-intensive, and latency-sensitive. OpenRTiST uses machine learning filtering to overlay various artistic styles onto video frames, creating an augmented reality experience. In this experiment, the OpenRTiST client connects to the server either at the edge, specifically the testbed servers, or in the cloud via a server located at Carnegie Mellon University (CMU), in Pittsburgh, Pennsylvania, through a COTS 5G system and the adv-03 node. In Figure 4, we depict the corresponding critical performance indicators such as frames per second (FPS) and round-trip time (RTT) measured using the testbed. In Figure 4a, we plot the Cumulative Distribution Function (CDF) of FPS, and in Figure 4a we plot the Complementary CDF (CCDF) of RTT. As we move from edge to cloud, RTT significantly increases which results in lower refresh rates and less responsive video stream. The performance is benchmarked against different video frame sizes to analyze network and compute capacity.

For the second claim, we concentrate on wireless network latency, a metric inherently challenging to model due to its stochastic nature at the packet level. While latency can be influenced by factors like transmit power and channel interference, it also depends on wireless protocol settings such as modulation schemes, bandwidth, and antenna configurations. ExPECA's isolated location minimizes variability in channel quality, mitigating potential interference. In Figure 5, we present latency measurements across different radio links over several days, demonstrating the reproducibility level in the case of COTS 5G wireless links. The close similarity between these CCDFs confirms the reproducibility of wireless latency measurements achieved through ExPECA.

4 SUPPORTED EXPERIMENTATION

This section delves into the features of the testbed that users can customize to meet their unique experimental goals. These features can be broadly categorized into two types: (1) configurations related to workload and (2) configurations related to the network.

In terms of workload-related configurations, ExPECA offers researchers a range of options to tailor to their experiments. Users can select from various geographical locations for computational workloads using the cloud interface and by reserving public IPs. It is possible to integrate GPUs for tasks requiring accelerated processing, or choose a CPU with a certain number of cores, and allocate a specific amount of RAM to benchmark workload performance.

¹Experiments available as Python Jupyter notebooks at <https://github.com/KTH-EXPECA/examples>

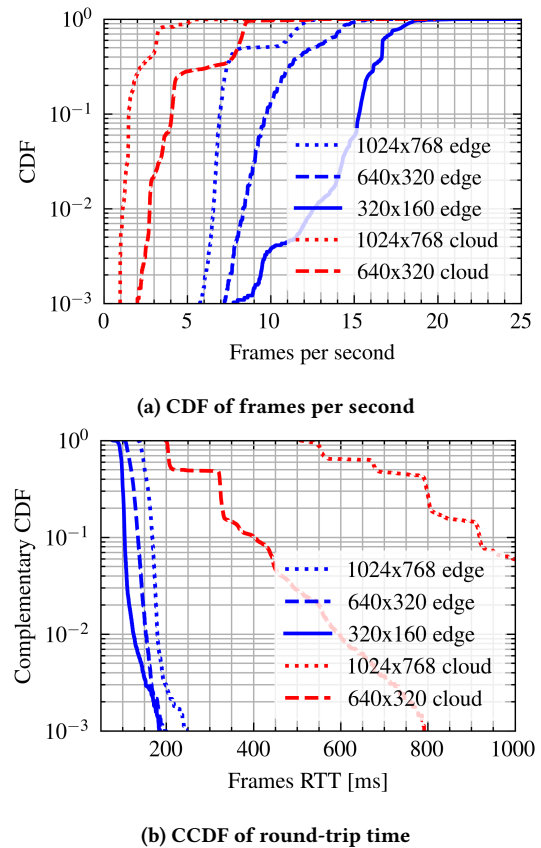


Figure 4: Performance analysis of OpenRTiST on COTS 5G system and adv-03 node using the ExPECA testbed.

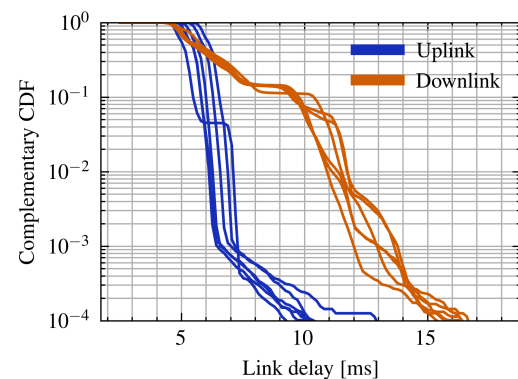


Figure 5: Twelve rounds of independent end-to-end latency measurements over the 5G COTS system between dot-02 and adv-03 node, each based on 160k samples.

The platform supports experiments involving both static (fixed-location) and mobile (varying-location, e.g., drone with a wireless connectivity dongle) wireless end nodes.

For network-related configurations, ExPECA provides a diverse array of network types, such as SDR 5G, SDR LTE, SDR WiFi, and

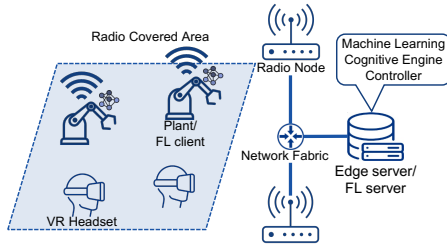


Figure 6: Illustration: Testbed experiment scenarios.

COTS 5G, among others. Researchers can also customize network topologies and control interference levels, thanks to the testbed’s isolated location. The platform supports a wide range of wireless protocols compatible with USRP e320 and allows users to configure channel conditions. Additionally, the use of Software Defined Radios (SDRs) offers further flexibility, including control over transmit power and the ability to implement a variety of wireless protocols.

ExPECA offers the opportunity to explore various experimentation scenarios involving edge computing, wireless communication, and machine learning research. Here we explain various scenarios, each shedding light on some of the interesting aspects of wireless communication and/or edge computing offering unique insights into the dynamic landscape of edge computing. In Figure 6, we provide an illustration that helps a reader to visualize these scenarios.

Below, we provide a few of such use cases in various research areas. While some of these aim at direct measurements for data-driven modeling and prediction, others target initial observations for forthcoming theoretical work or the verification and validation of existing theoretical models.

4.1 Optimization of Closed-Loop Applications

With the booming interest and applications surrounding cyber-physical systems, it is imperative to study Networked Control Systems (NCSs) [22]. NCSs form crucial control systems where the control loops are closed through a communication network. Due to the safety-critical nature of the applications these NCSs encompass, such as robots, UAVs etc., there is a need to benchmark and understand the bottlenecks/boundaries associated with these applications [15]. Namely, the latency and reliability boundaries of robotic control are crucial to understand before they can be deployed in real-world scenarios, where their potential failure might prove expensive [14].

Another such feedback system where latency is of importance are human-in-the-loop applications such as augmented reality, wearable cognitive assistants (WCA) [5, 15], and ambient safety. In the context of criticality, we have systems of automated fault detection, in which case the acoustic [9] or motion amplified visual [8, 13] data is processed for vibration analysis to potentially trigger some maintenance, safety or emergency procedures [9]. The ExPECA testbed provides developers with the opportunities to experiment with workloads corresponding to these types of applications in a containerized manner with various real network architectures. These experiments could either be for data collection or for verification of existing theoretical models.

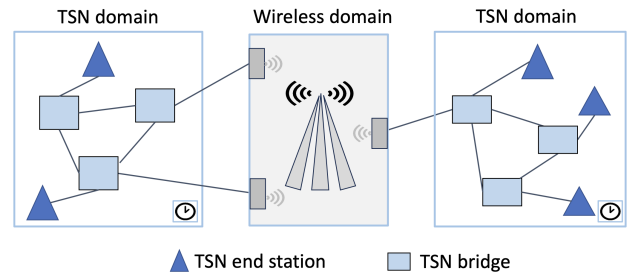


Figure 7: An illustration for the integration of a wireless domain with TSN domains.

4.2 Time-sensitive Wireless Networking

Supporting the above-mentioned CPS and HITL applications requires the network to offer a certain level of *end-to-end* network latency, often combined with extremely high reliability, in order to operate correctly and safely. This has spurred various efforts to integrate wireless networking technologies (e.g., 5G URLLC) with wired technologies (e.g., TSN) as shown in Figure 7. However, an efficient wired-wireless integration realization to support time-critical applications necessitates accurate system characterization. In particular, latency characterization of wireless segments is crucial as they are subject to various stochastic and temporal influences, unlike wired links where latencies are immune to external influences. Furthermore, accurate predictions of wireless latency are also envisioned to be leveraged in 6G networks to support deterministic communications [18].

Lately, several data-driven methods for latency prediction have been proposed [12, 19]. Training and validating these approaches require a vast amount of datasets comprising latency measurements alongside network parameters (e.g., SINR and MCS). Precise time-synchronization of all nodes of the testbed ensures accurate latency measurements can be collected between any two nodes of the testbed. These experiments can employ diverse wireless communication technologies, including COTS 5G, software-defined 5G and IEEE 802.11g Wi-Fi integrated in ExPECA. Furthermore, the containerized workflow within ExPECA enables users to perform reproducible and repetitive latency measurement experiments.

4.3 Hierarchical Learning and Inference

Another set of experiments researchers can use the ExPECA testbed for is the verification of theoretical results corresponding to Hierarchical Learning (HI) [1, 11]. In these works, authors develop the idea of Hierarchical Inference, where a compute constrained edge device chooses to perform local inference or classification of an image sample, or offloading via a wireless channel to get help of a high power edge server for classification. A small-sized ML model with limited accuracy is assumed to have been deployed at the edge device and offloading to the server incurs cost. In essence, these policies aim to make intelligent decisions regarding task offloading, prioritizing complex tasks or tasks where accuracy is paramount. Conversely, they process simpler tasks or tasks where reduced latency is of utmost importance locally. It is to be noted that in this specific work on Hierarchical Inference this decision is made in an

online manner. Using the ExPECA testbed, researchers can plan to implement the various algorithms proposed and verify if the real-world results match simulations and models.

4.4 Federated Learning Validation

Federated learning (FL) is emerging as one of the key approaches for training machine learning models [10]. FL aims to improve over centralized learning in terms of privacy, total training time, computation overhead, etc. However, FL can cause large communication overhead because the machine learning model should be repeatedly exchanged between the central server and the participating nodes during the training process. In order to analyze and optimize communication overhead, many works have proposed models for FL systems operating in wireless environments [3, 16]. Validating these theoretical models in real systems is required to reveal otherwise unavailable insights. The ExPECA testbed supports experimentation involving both centralized learning (CL) and federated learning to evaluate them in terms of communication/computation overhead and accuracy with various wireless conditions. For example in case of communication overhead, CL and FL will consume communication resources when exchanging data (i.e., using uplink to send input data in CL and uplink/downlink to send machine learning model in FL). The ExPECA testbed is able to measure the communication resource usage, and training time (communication/computation time), and capture wireless parameters at that time. Thanks to improved reproducibility and repeatability, the ExPECA testbed supplies stable validation for machine learning architectures.

5 ACKNOWLEDGEMENTS

This research has been partially funded by (1) the VINNOVA Competence Center for Trustworthy Edge Computing Systems and Applications (TECoSA) at KTH Royal Institute of Technology; and (2) the Swedish Foundation for Strategic Research (SSF), through grant number ITM17–0246.

REFERENCES

- [1] Ghina Al-Atat, Andrea Fresa, Adarsh Prasad Behera, Vishnu Narayanan Moothedath, James Gross, and Jaya Prakash Champati. 2023. The Case for Hierarchical Deep Learning Inference at the Network Edge. In *Proceedings of the 1st International Workshop on Networked AI Systems* (Helsinki, Finland) (*NetAISys '23*). Association for Computing Machinery, New York, NY, USA, Article 3, 6 pages. <https://doi.org/10.1145/3597062.3597278>
- [2] Joe Breen, Andrew Buffmire, Jonathon Duerig, Kevin Dutt, Eric Eide, Mike Hibler, David Johnson, Sneha Kumar Kaseria, Earl Lewis, Dustin Maas, Alex Orange, Neal Patwari, Daniel Reading, Robert Ricci, David Schurig, Leigh B. Stoller, Jacobus Van der Merwe, Kirk Webb, and Gary Wong. 2020. POWDER: Platform for Open Wireless Data-driven Experimental Research. In *Proceedings of the 14th International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization* (WiNTECH). <https://doi.org/10.1145/3411276.3412204>
- [3] Hongchang Gao, An Xu, and Heng Huang. 2021. On the Convergence of Communication-Efficient Local SGD for Federated Learning. *Proceedings of the AAAI Conference on Artificial Intelligence* 35, 9 (May 2021), 7510–7518. <https://doi.org/10.1609/aaai.v35i9.16920>
- [4] Shilpa George, Thomas Eisler, Roger Iyengar, Haithem Turki, Ziqiang Feng, Junjie Wang, Padmanabhan Pillai, and Mahadev Satyanarayanan. 2020. OpenRTiST: End-to-End Benchmarking for Edge Computing. *IEEE Pervasive Computing* 19, 4 (2020), 10–18. <https://doi.org/10.1109/MPRV.2020.3028781>
- [5] Kiryong Ha, Zhuo Chen, Wenlu Hu, Wolfgang Richter, Padmanabhan Pillai, and Mahadev Satyanarayanan. 2014. Towards Wearable Cognitive Assistance. In *Proc. of the 12th Annual International Conference on Mobile Systems, Applications, and Services*. Association for Computing Machinery, 68–81. <https://doi.org/10.1145/2594368.2594383>
- [6] Kate Keahey, Jason Anderson, Michael Sherman, Cody Hammock, Zhuo Zhen, Jenett Tillotson, Timothy Bargo, Lance Long, Taimoor Ul Islam, Sarath Babu, Hongwei Zhang, and François Halbach. 2022. CHI-in-a-Box: Reducing Operational Costs of Research Testbeds. In *Practice and Experience in Advanced Research Computing* (Boston, MA, USA) (*PEARC '22*). Association for Computing Machinery, New York, NY, USA, Article 14, 8 pages. <https://doi.org/10.1145/3491418.3530768>
- [7] Kate Keahey, Jason Anderson, Zhuo Zhen, Pierre Riteau, Paul Ruth, Dan Stanzione, Mert Cevik, Jacob Colleran, Haryadi S. Gunawi, Cody Hammock, Joe Mambretti, Alexander Barnes, François Halbach, Alex Rocha, and Joe Stubbs. 2020. Lessons Learned from the Chameleon Testbed. In *Proceedings of the 2020 USENIX Annual Technical Conference (USENIX ATC '20)*. USENIX Association.
- [8] Ce Liu, Antonio Torralba, William T Freeman, Frédo Durand, and Edward H Adelson. 2005. Motion magnification. *ACM transactions on graphics (TOG)* 24, 3 (2005), 519–526.
- [9] Bin Lu, Yaoyu Li, Xin Wu, and Zhongzhou Yang. 2009. A review of recent advances in wind turbine condition monitoring and fault diagnosis. In *IEEE Power Electronics and Machines in Wind Applications*. 1–7. <https://doi.org/10.1109/PEMWA.2009.5208325>
- [10] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 54)*. PMLR, 1273–1282. <https://proceedings.mlr.press/v54/mcmahan17a.html>
- [11] Vishnu Narayanan Moothedath, Jaya Prakash Champati, and James Gross. 2023. Online Algorithms for Hierarchical Inference in Deep Learning applications at the Edge. *arXiv:2304.00891* [cs.LG]
- [12] Samie Mostafavi, Gourav Prateek Sharma, and James Gross. 2023. Data-Driven Latency Probability Prediction for Wireless Networks: Focusing on Tail Probabilities. *arXiv preprint arXiv:2307.10648* (2023).
- [13] Tae-Hyun Oh, Ronnachai Jaroensri, Changil Kim, Mohamed Elgharib, Frédo Durand, William T Freeman, and Wojciech Matusik. 2018. Learning-based video motion magnification. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 633–648.
- [14] Manuel Olguin Muñoz, Neelabhro Roy, and James Gross. 2022. CLEAVE: Scalable and Edge-Native Benchmarking of Networked Control Systems. In *Proceedings of the 5th International Workshop on Edge Systems, Analytics and Networking* (Rennes, France) (*EdgeSys '22*). Association for Computing Machinery, New York, NY, USA, 37–42. <https://doi.org/10.1145/3517206.3526272>
- [15] Manuel Olguin Muñoz. 2023. *An Emulation-Based Performance Evaluation Methodology for Edge Computing and Latency Sensitive Applications*. Ph. D. Dissertation. KTH Royal Institute of Technology. <https://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-327222>
- [16] Jake Perazzone, Shiqiang Wang, Mingyue Ji, and Kevin S. Chan. 2022. Communication-Efficient Device Scheduling for Federated Learning Using Stochastic Optimization. In *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications*. 1449–1458. <https://doi.org/10.1109/INFOCOM48880.2022.9796818>
- [17] Dipankar Raychaudhuri, Ivan Seskar, Gil Zussman, Thanasis Korakis, Dan Kilper, Tingjun Chen, Jakub Kolodziejcki, Michael Sherman, Zoran Kostic, Xiaoxiong Gu, Harish Krishnaswamy, Sumit Maheshwari, Panagiotis Skrimponis, and Craig Gutterman. 2020. *Challenge: COSMOS: A City-Scale Programmable Testbed for Experimentation with Advanced Wireless*. Association for Computing Machinery, New York, NY, USA. <https://doi.org/10.1145/3372224.3380891>
- [18] Gourav Prateek Sharma, Dhruvin Patel, Joachim Sachs, Marilet De Andrade, Janos Farkas, Janos Harmatos, Balazs Varga, Hans-Peter Bernhard, Raheeb Muzaffar, Mahin K Atiq, et al. 2023. Towards Deterministic Communications in 6G Networks: State of the Art, Open Challenges and the Way Forward. *arXiv preprint arXiv:2304.01299* (2023).
- [19] Marco Skocaj, Francesca Conserva, Nicol Sarcone Grande, Andrea Orsi, Davide Micheli, Giorgio Ghinamo, Simone Bizzarri, and Roberto Verdona. 2023. Data-driven Predictive Latency for 5G: A Theoretical and Experimental Analysis Using Network Measurements. *arXiv preprint arXiv:2307.02329* (2023).
- [20] Jiakai Yu, Tingjun Chen, Craig Gutterman, Shengxiang Zhu, Gil Zussman, Ivan Seskar, and Daniel Kilper. 2019. COSMOS: Optical Architecture and Prototyping. In *2019 Optical Fiber Communications Conference and Exhibition (OFC)*. 1–3.
- [21] Hongwei Zhang, Yong Guan, Ahmed Kamal, Daji Qiao, Mai Zheng, Anish Arora, Ozdal Boyraz, Brian Cox, Thomas Daniels, Matthew Darr, et al. 2022. ARA: A Wireless Living Lab Vision for Smart and Connected Rural Communities. In *Proceedings of the 15th ACM Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*. 9–16.
- [22] Xian-Ming Zhang, Qing-Long Han, Xiaohua Ge, Derui Ding, Lei Ding, Dong Yue, and Chen Peng. 2020. Networked control systems: a survey of trends and techniques. *IEEE/CAA Journal of Automatica Sinica* 7, 1 (2020), 1–17. <https://doi.org/10.1109/JAS.2019.1911651>

Received 9 September 2023; accepted 26 September 2023