Privacy Preserving Federated Learning for Advanced Scientific Ecosystems

Rick Archibald*, Addi Malviya Thakur*, Marshall McDonnell*, Gregory Cage*, Cody Stiner*, Lance Drane*, Paul Laiu*, Michael J. Brim*, Mathieu Doucet†, William T. Heller† and Ryan Coffee‡.
*Computing and Computational Sciences Directorate
Email: {archibaldrk,malviyaa,mcdonnellmt,cagege,stinercc,dranelt,brimmj,laiump}@ornl.gov
†Neutron Sciences Directorate
Oak Ridge National Laboratory, Oak Ridge, TN
Email: {doucetm,hellerwt}@ornl.gov
‡ SLAC National Accelerator Laboratory, Menlo Park, CA
Email: coffee@slac.stanford.edu

2024 IEEE International Conference on Big Data (BigData) | USGov | DOI: 10.1109/BigData62323.2024.10825977

***Abstract*—We present a framework to provide privacy preserving (PP) federating learning (FL) across multiple computational and experimental facilities. This work joins the compute capabilities of National Energy Research Scientific Computing Center (NERSC) and Oak Ridge National Laboratory Research Cloud (ORC) with simulated experimental data, such as those produced at the SLAC National Accelerator Laboratory and Spallation Neutron Source (SNS). We describe the software infrastructure developed to provide privacy for computational and experimental networks. We developed algorithmic privacy across the federated system by embedding database security, computation, and communication into the federation architecture, utilizing scientific tools developed by the experimental community.**

***Index Terms*—Federated Machine Learning, Distributed Data, Privacy, Security, Scientific Ecosystems**

I. Introduction

The DOE has maintained leadership in computational and physical science over its history by continually building and upgrading leadership facilities that the world uses to further our scientific understanding. Figure 1 displays the locations of just a few of the large experimental scattering facilities run by the DOE. Traditionally, these facilities operated independently of each other because pushing the edge of scientific and computing facilities requires a focused approach. Over the last two decades the Scientific Discovery through Advanced Computing (SciDAC) program at the DOE [1] has been successful in bringing together diverse disciplines through partnerships of applied mathematicians, computer scientists, and scientists to deliver breakthrough scientific results. Recently, there has been an active drive to extend this collaborative approach to the facilities level. The Integrated Research Infrastructure (IRI) is a concerted effort by the DOE to seamlessly integrate research facilities across the DOE complex to create next generation super facilities [2]. This program is in the development phase, with an initial focus on developing the infrastructure for a connected scientific ecosystem. The diverse partnerships enabled by integrated scientific facilities will generate a new set of security and privacy challenges that must be solved to realize the full potential of this new paradigm.

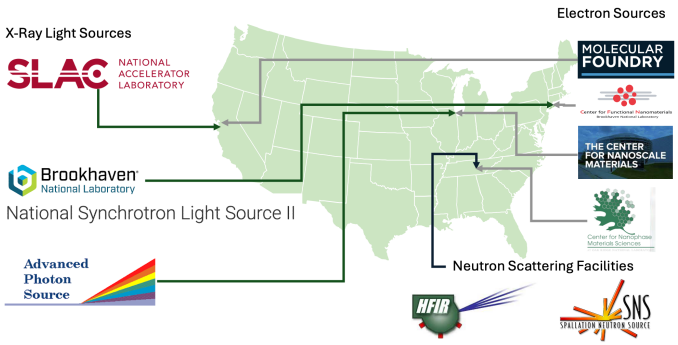

Fig. 1. DOE large experimental scattering facilities are distributed across the nation, with examples shown here. Movement of raw experimental data for co-analysis is expensive and there are security concerns.

There has been explosive growth in the volume of computational and scientific experiments performed at ever-increasing precision and resolution. Concurrently, similar growth is taking place in computing and networking, providing new opportunities to analyze large complex data sets using machine learning and artificial intelligence. Science is at unique crossroads, where integration of these advances with fundamental computer science and mathematical research has the potential to accelerate scientific discovery. Here, we specifically focus on the problem of securely linking large experimental facilities to large computational facilities that reside at different laboratories. Figure 2 depicts the PP-FL connecting three different large experimental scattering facilities discussed herein. The facilities provide three different scattering techniques, namely Electron, Neutron, X-ray scattering. Each of these probes produces complementary information about material properties. Methods that can synthesize the various results are needed to

979-8-3503-6248-0/24/$31.00 ©2024 IEEE

take full advantage of all the data produced at these facilities. We assume that experimental data can come in two different flavors, open science and secure science. The latter refers to science that may be classified for national security, be proprietary information, or require strong privacy protections for other reasons. This project integrates the control and data plane of INTERSECT [3], [4] for orchestrating the FL, uses security standards for authentication to ensure private access of data, includes network security for shared derivative information between FL sites, and embeds algorithmic security into the federated learning method used to learn and analyze distributed experimental data.

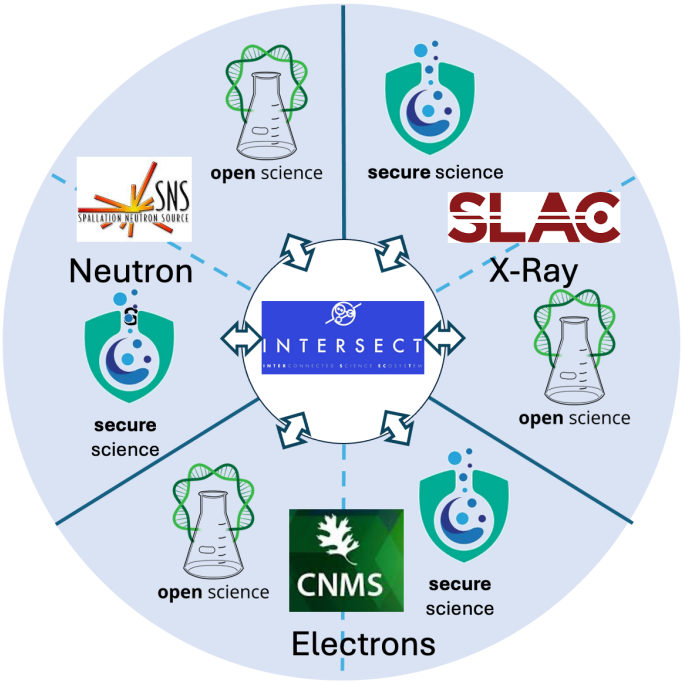

Fig. 2. Secure and privacy preserving demonstration on multimodal scattering data from three different facilities. Data at each facility is never moved and can be either open science or secure/private science information.

The paper describes a computer science and mathematical solution to enhance PP-FL across distributed experimental facilities. This paper also demonstrates the additional benefits that come from the design of our infrastructure. First, this framework can securely operate on any combination of distributed, secure, and open data. This framework is generalizable to arbitrary data formats and hardware setups. Second, this framework allows for dynamically adding / removing data sources during federated learning. Third, our algorithmic security does not move the experimental data from its original location or generation point. Instead, it only moves sparse, encrypted information derived from the data. By using such sparse information, there is an algorithmic benefit to privacy provided by the ability to encrypt the sparsity pattern. The subsequent reduction of network communications improves energy efficiency as well. Currently, this energy efficiency is bounded by the compression ratio of the lossless compression of ML parameters. On going work is being done to fully identify benefits, both to computation and communication of more advance scientific compresion methods in this framework.

II. Federated Privacy Preserving Framework

The DOE's AI for Science reports [5], [6] outline the need for intelligent systems, instruments, and facilities to enable scientific breakthroughs with autonomous experiments, self-driving laboratories, smart manufacturing, and AI-driven design, discovery, and evaluation. The DOE's Computational Facilities Research Workshop report [7] identifies intelligent systems/facilities as a challenge with enabling automation and eliminating human-in-the-loop needs as a crosscutting theme. The national science and technology council of the president released a report outlining the national strategy to advance Privacy-Preserving Data Sharing and Analytics (PPDSA), where they identify FL as one of eight PPDSA technology [8]. Together, these reports affirm that PP-FL will play an important role in the future of the DOE complex.

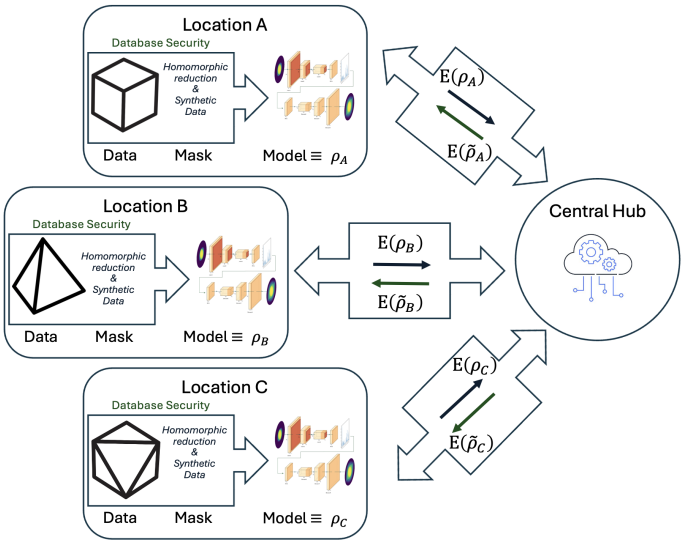

Fig. 3. Leftmost boxes are an abstraction of the distributed locations in a FL architecture. Individual machine learners at each location operating on local data, where only machine learning parameters are transferred during the training phase to centralized location or aggregator. Privacy preservation can be increased through encryption of both data and communicated parameters.

To achieve the goals of enhancing Federate Learning abilities as a PPDSA technology, we will bolster through computer science and mathematical advancements into a common framework for scientific discovery. To this end, we considered the privacy preserving methods in both the scientific/computational ecosystems and embedded in algorithms below.

Scientific/Computational Ecosystems.

- Build on the successes of ORNL's Interconnected Science Ecosystem (INTERSECT) federated software for securely and efficiently executing computation on remote resources distributed across facilities.

- Co-design our ML algorithms to seamlessly interface with INTERSECT using algorithms that balance computational loads across networked computing resources, providing users with direct access to information and methods developed in the greater scientific community.

Algorithmic Security and Privacy.

- Algorithmically enhance FL by embedding other key PPDSA technologies into the learning architecture.
- Develop algorithmic database security across the different FL locations by using the technologies of differential privacy, synthetic data, and homomorphic reduction encryption.

The framework of the FL used is presented in Figure 3. FL arose from the need to train machine learning algorithms on data distributed across multiple locations without moving all of the data to a single physical location. FL is a PPDSA technology because analysis can occur on distributed data without having to risk transmitting data. FL only transmits the machine learning parameters derived from the data across the network, which provides a layer of security that has clear limits. If an adversary intercepts the transmitted data, there are known methods for reverse engineering the training data from the machine learning parameters know as model inversion attacks [12]. The goal of our developed approach was to strengthen the security and privacy of FL by algorithmic privacy protection on each dataset locally and then adding algorithmic security across the network. The accuracy of our proposed FL algorithm was tested against the gold standard for testing FL algorithms, which is to compare against a single learner with access to all distributed data. The details of our INTERSECT solution are given schematically in Figure 4.

The PP methods employed, specifically synthetic data, homomorphic computing, and their costs are listed in Table I. Users are informed on the cost of each PP method and can opt for any combination of these methods. A schematic of the integration of these methods with network encryption is displayed in Figure 3. This approach makes synthetic data techniques straightforward and effective. Since our data is generated by SasView simulation [13], it is possible to generate completely new data using the same statistical distributions to achieve synthetic PP in our FL framework. We achieve homomorphic computing using the privacy preserving package CrypTen (version 0.4.0) [14], [15], where we perform machine learning and prediction on encrypted data, producing an encrypted answerer. Only users with the encryption key and encryption method are able to use the machine learning algorithm to produce viable results. The last type of PP methods used is network encryption, where the package pyAesCrypt (version 6.1.1) is used to encrypt and decrypt network parameter streams in the AES encryption format [16]. We use the federated algorithm proposed in [17], with out low rank approximation, that has been demonstrated to be robust and accelerated method for FL training. The focus of this manuscript is to present PP-FL in advanced scientific ecosystems, for a recent review of advances in FL and PP we refer the reader to the review article [18].

III. Numerical Results

The implementation of PP-FL presented here was demonstrated with small angle scattering (SAS) data, which is a widely-employed materials characterization technique implemented at the DOE X-ray and neutron scattering facilities and elsewhere. SAS probes structures at length scales ranging from 1 to 100 nanometers. The SAS community has produced Sasview [13] a data analysis software tool adopted throughout the international user community. Here, Sasview (version 5.0.6) was used to simulate a large distributed database of SAS data. Simulated SAS data was generated for forty different models used in SAS data analysis [13], which covers most of the models implemented but excludes those that are computationally expensive. Models in Sasview are mathematical functions that describe the SAS signal that would be observed from the incredibly diverse kinds of material structures that can be characterized by the technique. Each model is parameterized through a set of physical parameters that can take on a wide range of values. Data were generated for each model 6K times from a random uniform sampling over the physically relevant parameter space, as described in previous work [19]. Once generated, the data were split into six equally-sized, distributed databases for the FL based on clustering the values in the parameter space. The approach creates heterogeneous databases that mimic the heterogeneity of the databases of measured experimental data found in the facilities across the DOE complex, akin to diversity of data indicated in Figure 2. We split the distributed databases into 90% training, 9% testing, and 1% validation sets. Note that only the training data have the heterogeneous distribution introduced by clustering, whereas the testing and validation sets maintain a homogeneous distribution across the whole sampled modeled space of SasView.

A standard convolutional auto encoder was implemented for local learners, as shown in Figure 5. We note our FL framework is agnostic to the particular ML methods utilized as long as each datasite ML method is similar. There are three convolutional/pooling steps, denoted by **CN** and a two layer fully-connected network, denoted by **FCN**. These are followed by three deconvolutional steps, denoted by **DN**. In SAS, the two-dimensional intensity function, I, is measured. The data can be provided as a two dimensional function in cylindrical coordinates of the angle around the beam direction, ϕ, and the angle from the beam direction that is measured as the radius away from the center of the pattern, r. Given this $I(r, \phi)$ as the ground truth, the data are then convoluted with the pinhole smearing function accessible in Sasview to represent instrumental effects in a real measurement and add Poisson noise to approximate the measurement uncertainty. The autoencoder is trained to learn the decovolution and denoising transformation that takes our simulated instrument measurements to simulated ground truth.

In order to facilitate the training of this data across multiple locations, the INTERSECT-SDK framework [4] is utilized to

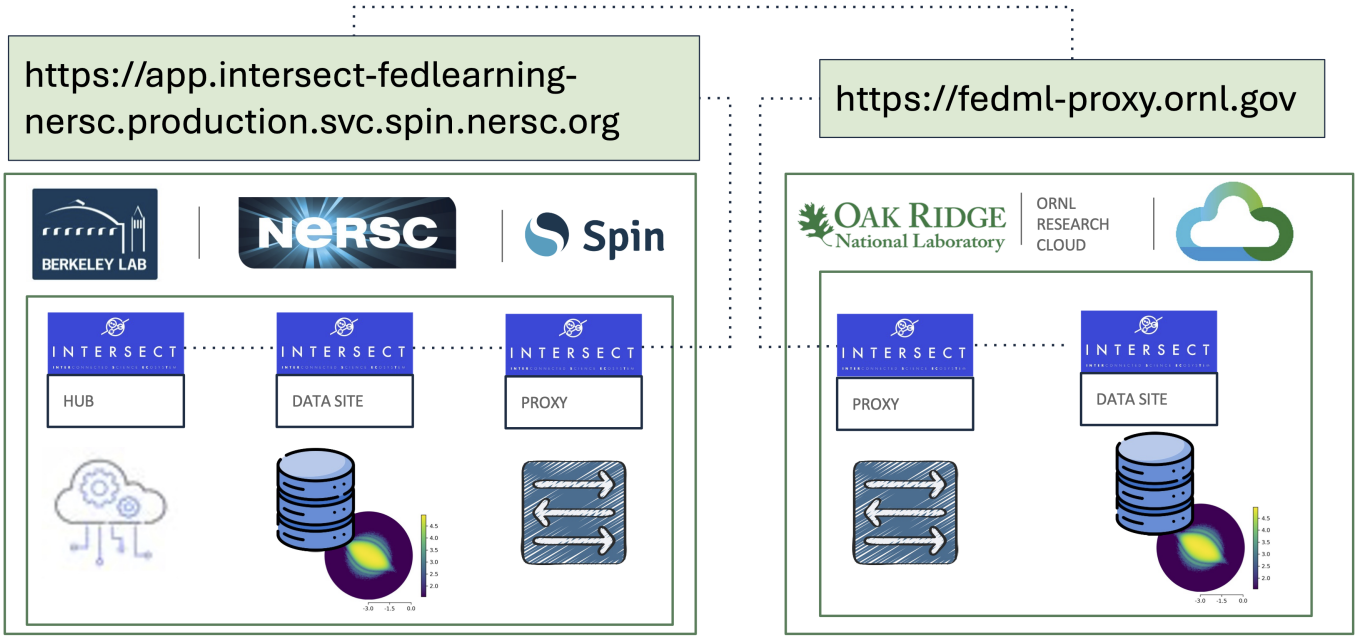

Fig. 4. Design for performing FL across LBNL and ORNL using INTERSECT. The INTERSECT proxy connects two separate INTERSECT ecosystems where the data sites are distributed between both. These INTERSECT ecosystems were run at LBNL NERSC's Spin [9], [10] and ORNL's ORC. The FL hub is shown at LBNL (equally could be at ORNL). Domain names are shown for the INTERSECT proxies that are used for current developments. Icons used created by Juicy Fish Flaticon [11]

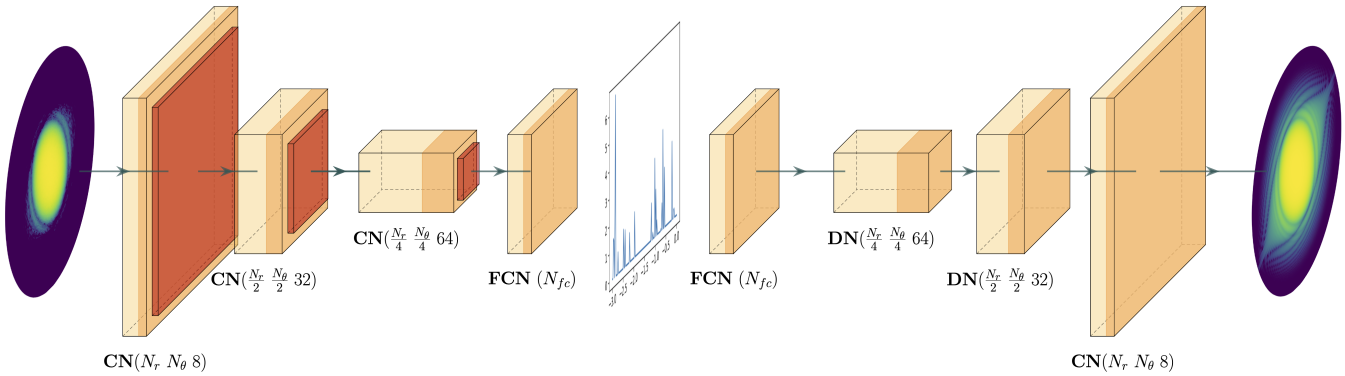

Fig. 5. Convolutional autoencoder architecture that deconvolves and denoises SAS data. There are three convolutional and decovolutional layers with a fully connected network of size N_{fc} in between. Here, N_r is the number of radial points, N_θ is the number of angular points, CN denotes a convolutional layer, FCN a fully connected layer, and DN a deconvolutional layer.

provide an ecosystem to connect the various locations and enable the sharing of data parameters. Each individual location supports a "FedSite", an Intersect service that houses the execution of the FL algorithms. Each FedSite has access to its location's training data and has externally available endpoints to start and end the training process. To connect all of the sites, a "FedHub" is created and hosted on one of the locations. The FedHub can "register" FedSites, allowing the hub to manage the training processes of each site. In the future, multiple FedHubs could potentially be set up to communicate with different subsets of locations, each of which could individually manage different rounds of learning.

Figure 6 showcases a demonstration of the INTERSECT FL ecosystem and how communication between the sites and the hub occurs. Initially, for a given FedHub, the sites used for a federated model training are registered with the FedHub, represented by blue arrows. The FedHub then begins training by sending a message to each site to commence training, and then waits for response from each site. Once all sites have responded with their updated parameters, the FedHub selects the data with the lowest error, and then sends those parameters back to each site for the next iteration of training. After each iteration of the training the FedHub also sends out a status update in the form of an INTERSECT-SDK event, with the

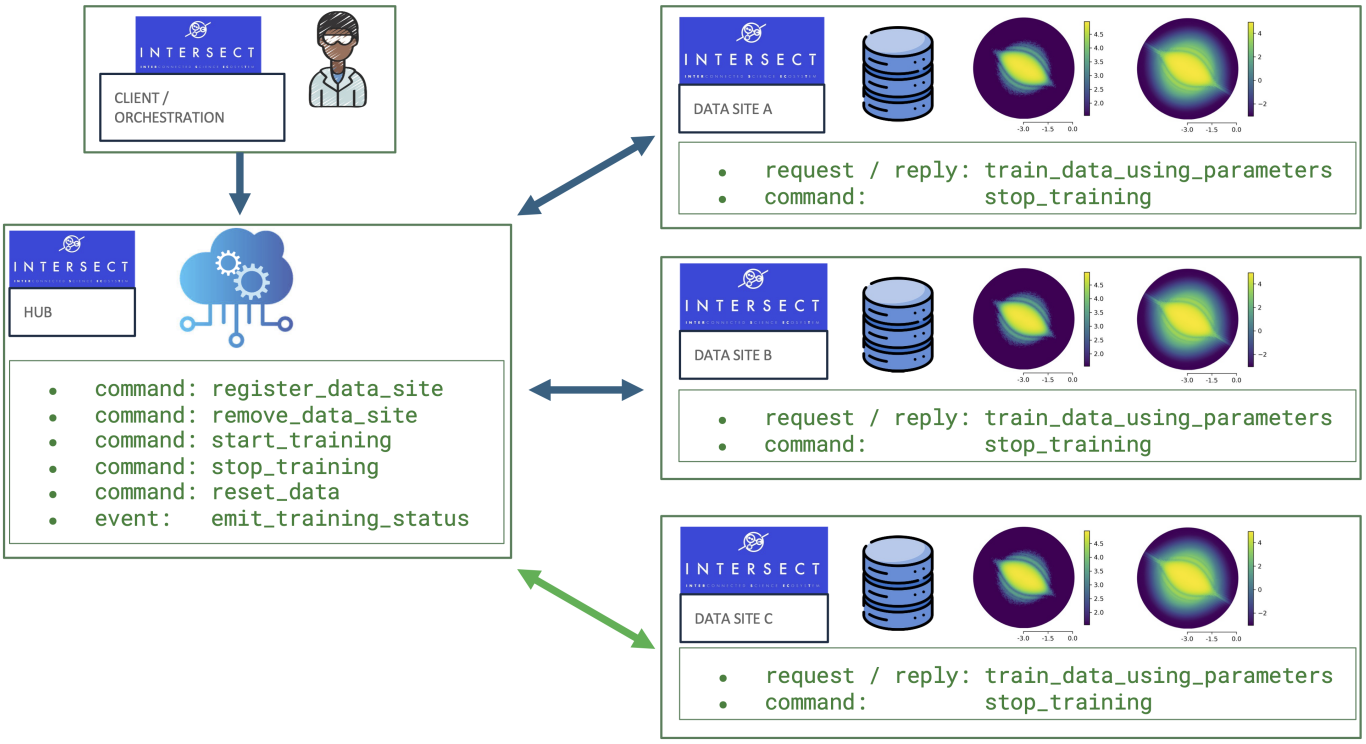

Fig. 6. Implementation of multi-site workflow for dynamically adding data sites to federated learning via INTERSECT. 1) the client / orchestration agent registers multiple data sites with the hub, as represented by the blue arrows. 2) Hub initiates training to the hub, which sends the initial training parameters to the registered data sites. After local training at the data sites is complete, 3) the data sites send back their results to the hub; after the hub receives all registered data sites' results, it outputs an event with the training status. The cycle of 2) and 3) will continue as long as the training is active. During this cycle of training, 4) the client / orchestration agent may add a new data site dynamically into the workflow with the hub, as represented by the green arrow, which will send the current parameter set to the new data site and 5) await for all registered sites to return their results, where the cycle of 2) and 3) can contiune with the new data site. Eventually, 6) the training is stopped via a command to the hub (initiated by a client or an orchestration agent due to the objective being achieved).

latest parameters.

A FedHub also has the capability to register and remove sites while in the middle of training. This enables the adding of a site to begin training after other sites have already started, represented by the green arrow. The hub sends the latest parameters to that new site, and the site begins training immediately. The hub will then monitor that site for its data and treat it the same as the rest of the sites. Removing a site behaves in the opposite manner. The site to be removed will stop training and FedHub will no longer collect the parameters from that site.

This framework was demonstrated using an orchestrator written to communicate with a FedHub running on a local machine. The orchestrator's workflow registers five FedSites also running locally and begins training via the FedHub. Each time an iteration of learning passes, FedHub emits an event detailing the current parameters that it is keeping track of. The orchestrator listens for this event and manages the lifecycle of the training based on how many status messages it has received. After some iterations have passed, the orchestrator registers another FedSite with the FedHub, bringing the total number of sites to six. The new site then begins the training process. After several more iterations, one of the originally registered sites is removed and the FedHub no longer oversees that site's training. Finally, after the orchestrator has received a configured amount of status messages, it communicates with the hub to stop the training.

Figure 7 displays a representative result for our FL methods. It can be seen that FL provides a ${}^{1.37e-03}/_{4.68e-04} \approx 3$ times improvement in error and is within $100 \times {}^{(4.68e-04-4.63e-04)}/_{4.63e-04} \approx 1\%$ relative difference in error with the gold standard, where the same method is trained on the data in a universal location.

Table I displays the computation cost for including different PP methods in FL. Note that the PP methods did not significantly affect the accuracy of the prediction, and so the results are not reported. Importantly, the cost of the PP methods are negligible for all but the prediction using homomorphic computation. The relative cost of homomorphic computing PP in prediction is high because, homomorphic computing incurs a major cost when the data is encrypted and decrypted. This cost is small compared to the cost of training but running the prediction on the test data is on the order of cost in comparison to the encryption. Both are relatively fast in comparison to the training, but could become burdensome for large numbers of predictions. Finally, synthetic data is not

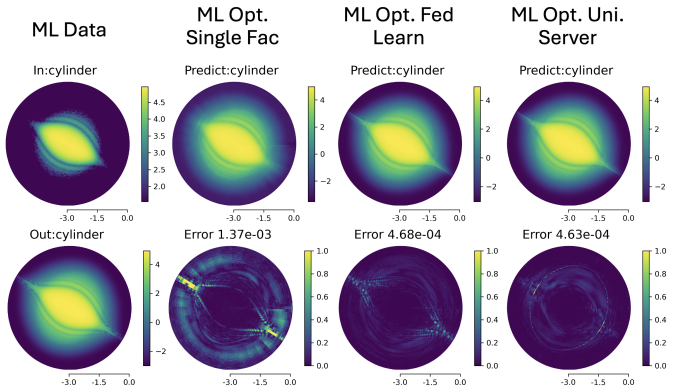

Fig. 7. Representation of accuracy of federated learning on SAS data. Bottom left is a representative testing sample of the cylinder model from SasView depicting the measured intensity. The top left includes pinhole smearing and Poisson noise, and then the next three pictures depict the best result across all distributed databases without federated learning, the result with federated learning, and finally the gold-standard result of training a single convolution auto encoder on all data from every database. The bottom rows below the predicted results show the error in the prediction.

needed to be generated for prediction, so cost is not a factor for this category. Similarly, the required prediction is done locally so network encryption is not needed for prediction and is not a factor in the overall cost.

TABLE I
COST IN TRAINING AND PREDICTION OF PP METHODS IN FL.

PPDSA Method	Training	Prediction
Synthetic Data	$1.2 \pm 3\%$	–
Homomorphic Computing	$2.8 \pm 0.2\%$	$226 \pm 9\%$
Network Encryption	$1.2 \pm 0.1\%$	–

IV. CONCLUSION

The project's primary goal was to develop a PP-FL framework for connecting facilities to enable researchers to leverage the potential of integrated scientific ecosystems. By enabling a seamless and PP sharing of scientific data among distributed facilities, we have demonstrated that the denoising and deconvolution of SAS data can improve. The development can foster a more collaborative research environment, supporting the DOE's transition toward an integrated network of smart laboratories. In addition, the energy-efficient and scalable nature of the proposed federated learning framework underscores its potential for broad applicability and sustainability within the scientific community, paving the way for future advancements in secure and efficient data analysis.

ACKNOWLEDGMENT

This work was supported by the Office of Advanced Scientific Computing Research and performed at the Oak Ridge National Laboratory, which is managed by UT-Battelle, LLC for the US Department of Energy under Contract No. DE-AC05-00OR22725. A portion of this research] used resources at the Spallation Neutron Source, a DOE Office of Science User Facility operated by the Oak Ridge National Laboratory. This work benefited from the use of the SasView application, originally developed under NSF award DMR-0520547. SasView contains code developed with funding from the European Union's Horizon 2020 research and innovation programme under the SINE2020 project, grant agreement No 654000. This research used resources from the ORNL Research Cloud Infrastructure at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725

REFERENCES

[1] Department of Energy, "Scientific discovery through advanced computing," www.scidac.gov, accessed:9/2024.
[2] W. L. Miller, D. Bard, A. Boehnlein, K. Fagnan, C. Guok, E. Lançon, S. J. Ramprakash, M. Shankar, N. Schwarz, and B. L. Brown, "Integrated research infrastructure architecture blueprint activity (final report 2023)," US Department of Energy (USDOE), Washington, DC (United States). Office of ..., Tech. Rep., 2023.
[3] Oak Ridge National Laboratory, "Interconnected science ecosystem (intersect)," www.ornl.gov/intersect, accessed:9/2024.
[4] GitHub, "INTERSECT-SDK GitHub Group," https://github.com/INTERSECT-SDK, accessed:9/2024.
[5] J. Carter, J. Feddema, D. Kothe, R. Neely, J. Pruet, R. Stevens, P. Balaprakash, P. Beckman, I. Foster, K. Iskra *et al.*, "Advanced research directions on ai for science, energy, and security: Report on summer 2022 workshops," Argonne National Laboratory (ANL), Argonne, IL (United States), Tech. Rep., 2023.
[6] R. Stevens, V. Taylor, J. Nichols, A. B. Maccabe, K. Yelick, and D. Brown, "Ai for science: Report on the department of energy (doe) town halls on artificial intelligence (ai) for science," Argonne National Lab.(ANL), Argonne, IL (United States), Tech. Rep., 2020.
[7] N. None, "Doe national laboratories' computational facilities-research workshop report," Argonne National Lab.(ANL), Argonne, IL (United States), Tech. Rep., 2020.
[8] Executive Office of the President of the United States, "National strategy to advance privacy-preserving data sharing and analytics. washington, dc,: National science and technology council," www.whitehouse.gov/wp-content/uploads/2023/03/National-Strategy-to-Advance-Privacy-Preserving-Data-Sharing-and-Analytics, 2023.
[9] LBNL, "Nersc spin," https://www.nersc.gov/systems/spin/.
[10] B. Enders, D. Bard, C. Snavely, L. Gerhardt, J. Lee, B. Totzke, K. Antypas, S. Byna, R. Cheema, S. Cholia, M. Day, A. Gaur, A. Greiner, T. Groves, M. Kiran, Q. Koziol, K. Rowland, C. Samuel, A. Selvarajan, A. Sim, D. Skinner, R. Thomas, and G. Torok, "Cross-facility science with the superfacility project at lbnl," in *2020 IEEE/ACM 2nd Annual Workshop on Extreme-scale Experiment-in-the-Loop Computing (XLOOP)*. IEEE, Nov. 2020. [Online]. Available: http://dx.doi.org/10.1109/xloop51963.2020.00006
[11] J. Fish, "FlatIcon Website," https://www.flaticon.com/authors/juicy-fish/sketchy, 2024.
[12] J. Song and D. Namiot, "A survey of the implementations of model inversion attacks," in *International Conference on Distributed Computer and Communication Networks*. Springer, 2022, pp. 3–16.
[13] S. project, "sasview.org," 2024.
[14] B. Knott, S. Venkataraman, A. Hannun, S. Sengupta, M. Ibrahim, and L. van der Maaten, "Crypten: Secure multi-party computation meets machine learning," in *arXiv 2109.00984*, 2021.
[15] Facebook Open Source, "A research tool for secure machine learning in pytorch," crypten.ai, accessed:9/2024.
[16] PyPI, "pyaescrypt 6.1.1," pypi.org/project/pyAesCrypt, accessed:9/2024.
[17] S. Schotthöfer and M. P. Laiu, "Federated dynamical low-rank training with global loss convergence guarantees," *arXiv preprint arXiv:2406.17887*, 2024.
[18] M. Aggarwal, V. Khullar, and N. Goyal, "A comprehensive review of federated learning: Methods, applications, and challenges in privacy-preserving collaborative model training," *Applied Data Science and Smart Systems*, pp. 570–575, 2024.

[19] R. K. Archibald, M. Doucet, T. Johnston, S. R. Young, E. Yang, and W. T. Heller, "Classifying and analyzing small-angle scattering data using weighted *k* nearest neighbors machine learning techniques," *Journal of Applied Crystallography*, vol. 53, no. 2, pp. 326–334, Apr 2020. [Online]. Available: https://doi.org/10.1107/S1600576720000552