Pacific Northwest
NATIONAL LABORATORY

PNNL- 36393

Homomorphic Encryption for Electrical Metering Aggregation

Protecting the Privacy of Building Tenants

August 2024

Syd Burtner

U.S. DEPARTMENT OF ENERGY

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

DISCLAIMER

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from
the Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062
www.osti.gov
ph: (865) 576-8401
fox: (865) 576-5728
email: reports@osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
or (703) 605-6000
email: info@ntis.gov
Online ordering: http://www.ntis.gov

PNNL- 36393

Homomorphic Encryption for Electrical Metering Aggregation

Protecting the Privacy of Building Tenants

August 2024

Syd Burtner

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Homomorphic Encryption for Electrical Metering Aggregation: Protecting the Privacy of Building Tenants

Keywords:

Privacy, data aggregation, homomorphic encryption, smart metering, data analysis

---

Electrical meters are devices that measure consumer electricity usage. The data collected by these meters is necessary for utility billing and electrical grid management but can also be used to assess the environmental impact of buildings. Prior research has found that unprotected metering data could potentially be used to infer some information about the behaviors of building tenants by detecting changes in electricity usage. For example, a period of low electricity usage could suggest that the tenants are not in the building. As smart metering becomes more common, there is a growing need for data privacy protections for metering data that do not negatively impact the quality and availability of data used for energy management and billing applications. To identify potential solutions, we developed a Python-based data aggregation platform to analyze the potential efficacy of privacy-enhancing technologies for energy metering applications. This platform aggregates groups of metering sites into virtual buildings, which could potentially detach changes in electrical activity from individual tenants, making it more difficult to track the activity of a specific tenant. To further protect data during analysis, this project utilizes homomorphic encryption as part of its initial approach. Homomorphic encryption offers a means of protecting energy consumption data while permitting mathematical operations to be performed without the need to know the data contents. This allows for data to be processed into usable statistics without revealing energy consumption information. A series of homomorphic encryption libraries were evaluated to determine their applicability and limitations in the context of metering data. The use of these techniques may help to reassure consumers and encourage further adoption of smart grid infrastructure.

Introduction

Electrical metering data is collected by utility companies for the purposes of billing customers, balancing the power grid, and enabling improvements to the environmental impact of buildings. As smart devices are increasingly integrated into the grid, more metering data has become available for these purposes, allowing for improved insight into electricity demand patterns (1). However, unprotected metering data could potentially be used to infer the behaviors of building tenants (2). There is thus a need to protect the privacy of electrical metering data.

Previous efforts to ensure the privacy of building tenants have not always accounted for the need to keep this data usable, limiting the applicability of these privacy protections (3). Data aggregation using homomorphic encryption offers one potential method of affording privacy to building tenants while providing usable metrics of electricity demand for billing and building improvement applications. Data aggregation is the process of combining multiple data sources into a single statistic, which may make it more difficult to trace a change in that data back to a specific source (4), while homomorphic encryption allows for mathematical operations to be performed on data without revealing the data to the party performing the calculations (5). This project seeks to create Python software to model the applicability and limitations of privacy protections for metering data. Its initial approach implements homomorphic encryption to provide data privacy protections during data aggregation.

Progress

During my internship under the Community College Internship program, I was tasked with creating a Python program to aggregate and analyze electrical metering data while protecting the privacy of that data with homomorphic encryption. Homomorphic encryption is a means of transforming data into an unrecognizable form while still allowing mathematical operations to be performed on that data. This means that statistics can be calculated without revealing the original data to the analyst (5). Data was homomorphically encrypted after aggregation and stored in its encrypted form, which protects against unauthorized viewing if an intruder gained access to the files while allowing those files to be used to calculate statistics.

The Northwest Energy Efficiency Alliance's dataset was used to test the efficacy of the program on metering data. This dataset contains up to 220,000,000 records per year of data (6), so calculation speed was a major consideration when designing a program to analyze it. We tested several Python libraries and algorithms for homomorphic encryption to determine their practicality for encrypting large quantities of metering data. Fully homomorphic encryption is capable of repeated additions, but it requires a procedure called bootstrapping to be performed at regular intervals to ensure that data remains accurate (7). Bootstrapping is effective but has historically been a computationally intensive process for many algorithms (8), which can extend processing times. After a technical review of encryption methods, partially homomorphic encryption was chosen because of its high speed of calculation and support for an unlimited number of additive operations without the need for bootstrapping (5).

Key size was another consideration. Encryption algorithms use numbers called encryption keys to transform data, and the recommended length of these keys varies between algorithms. Many algorithms require very large keys for long-term security, but the Elliptic Curve El-Gamal algorithm requires a much smaller encryption key size for long-term security; a key size of 521 bits is recommended by the National Institute of Standards and Technology, whereas exponential algorithms have a recommended key size of 15,360 bits (9). Using a smaller key size can save disk space, reduce data transmission overheads, and enable faster encryption and decryption of data (10) (11). The capabilities and small key size of the Elliptic Curve El-Gamal algorithm led to its selection for this project. It was also available in the library selected for this project. Of the libraries tested, the LightPHE library (12) was chosen for its exceptional speed while adding values, as aggregating the amount of data present in the Northwest Energy Efficiency Alliance's dataset required a substantial number of addition operations. LightPHE's addition times required less than one second per operation, while other libraries tested took up to four seconds per operation.

Processing speed remained a significant challenge throughout the project. The Northwest Energy Efficiency Alliance dataset consists of 323 metering sites, each of which can have several metered circuits (6). Each circuit logs meter readings at 15-minute intervals, resulting in 12-15 gigabytes of storage space per year of data. We significantly pared down the number of readings to process by limiting computation to the main circuit of each site, as this circuit aggregated the readings of all other circuits for that site. This narrowed the target to approximately 2 gigabytes of metering data per year. Completing data processing within the time bounds of the project required heavy optimization, and I learned to use the multiprocessing and multithreading Python libraries while reducing the runtime of this program. By using multiprocessing and optimizing program logic, we were able to process one year of data in approximately 1.5 hours, which was a significant improvement over the estimated pre-optimization time of approximately 24 hours. Further optimization consisted of separating data preparation from data analysis by saving prepared data to files, allowing for aggregation to take place at a separate time from data analysis. This means that the majority of the 1.5-hour runtime only needs to occur once per year of data.

The program's approach to preparing data for analysis consists of combining randomized groups of sites into virtual buildings. A default group size of 8 sites was chosen based off the results of a 2014 study done by the Pacific Northwest National Laboratory that found that the cumulative privacy benefits of increasing group size appear to plateau near a group size of 8 (4). To account for this possibility of a larger or smaller group size being desired, the program accepts an optional building size value; however, this value must be checked to ensure that it is large enough to provide privacy benefits. A value that is too small may not adequately obscure the energy demand of individual sites. A minimum building size of 4 buildings was chosen from the commercial requirements for aggregation in the United States (13), and the 2014 Pacific Northwest National Laboratory study agrees that an aggregation size of 4 or more provides reasonable privacy protection for individual meter readings (4). To ensure that sites are grouped

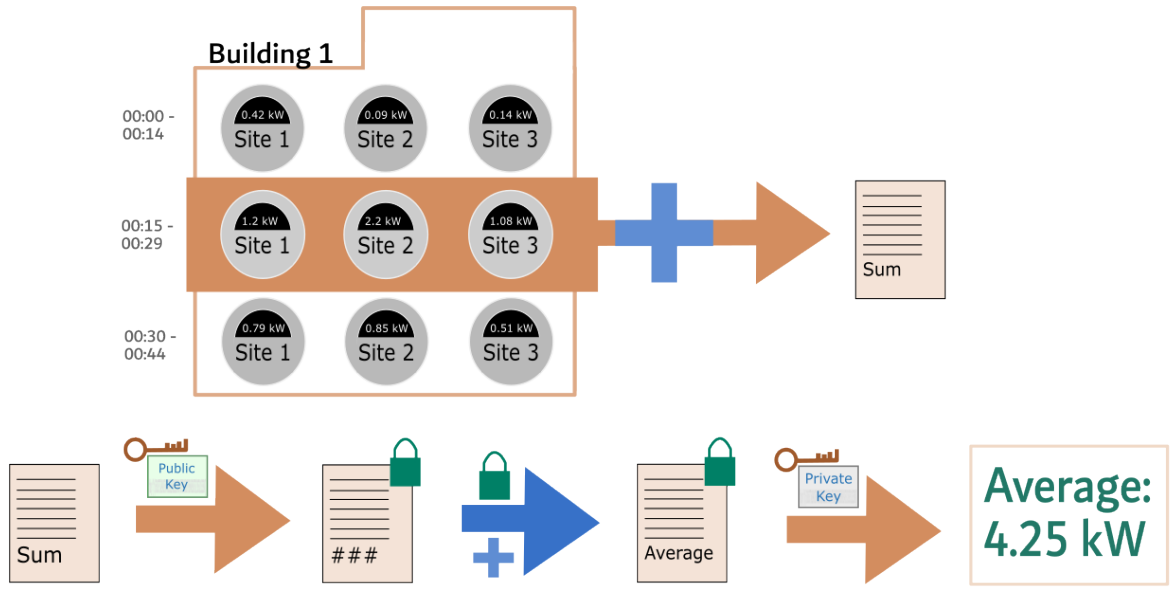

Figure 1: a visualization of the aggregation and analysis procedure.

the same way each time, a seed value is taken as input. This value is used to initialize a pseudorandom number generator so that it selects the same random groupings for each seed value, ensuring that sites are consistently grouped into the same buildings when analyzing multiple days of data. Further improvements to this algorithm are in progress and consist of choosing variable site counts to mirror real-world distributions of building sizes.

Aggregation consists of adding the electricity demand from each site into a composite value for each 15-minute interval. These 15-minute aggregates constitute the metering data for one virtual building. Performing this aggregation proved to be another challenge. Some meters did not report readings consistently, leaving missing readings in the data. The method used to add data together required that every 15-minute interval had a reading, so it was necessary to fill in these missing values. This process of filling missing data is called imputation (14). Imputing data required learning more about how pandas, a Python data analysis and processing library, handles data indexing, as well as learning about different methods of imputation.

The complete removal of missing records is common for data analysis, but doing so may bias the calculation of statistics if data is not missing completely at random (15). For the purposes of this program, forward-fill imputation was chosen and implemented. This method propagates forward the last existing value to fill missing data. Metering failures do not necessarily mean that a site has no electricity demand, and as energy demand is time-dependent, filling with the most recent value may provide a reasonable approximation of demand at that time. Forward-fill imputation is also natively supported by the pandas library, making it relatively simple to implement. This method is not perfect, however, as extended metering outages may not provide enough recent values to impute. In these cases, the use of historical data may be considered where possible to provide more relevant values based on prior observations for similar days. The implementation of historical data-based imputation relies on the existence

of similar historical data, however, and the presence of that data requires prior data processing that may not have taken place at runtime. As such, forward-fill imputation is the only method currently implemented. Further work on this project may implement imputation using historical data if present.

Once created, buildings are homomorphically encrypted and stored to file. The second module created for this program can read and decrypt data for virtual buildings from these files. It is also capable of basic data analysis; for instance, it can calculate the average energy demand for a given day. Decryption initially posed more of a challenge than anticipated, as the library used for encryption and decryption expected encrypted data in a specific format. The format output by the library used to read the encrypted data files was not the same as this expected format, so the decryption module needed to correct the formatting before decrypting data. Once this was done, decryption could be performed successfully, and we compared decrypted data with the original data to ensure that accuracy was maintained. Data consistently remained accurate to four decimal places, which is an acceptable level of accuracy. Several graphs were produced to demonstrate the calculation of average and total energy demand for a building within a day and year of time. Figures 2 and 3 demonstrate the successful calculation of total energy usage for a day and year, and Figure 3 demonstrates that the total and average energy consumptions appear as expected for the year.

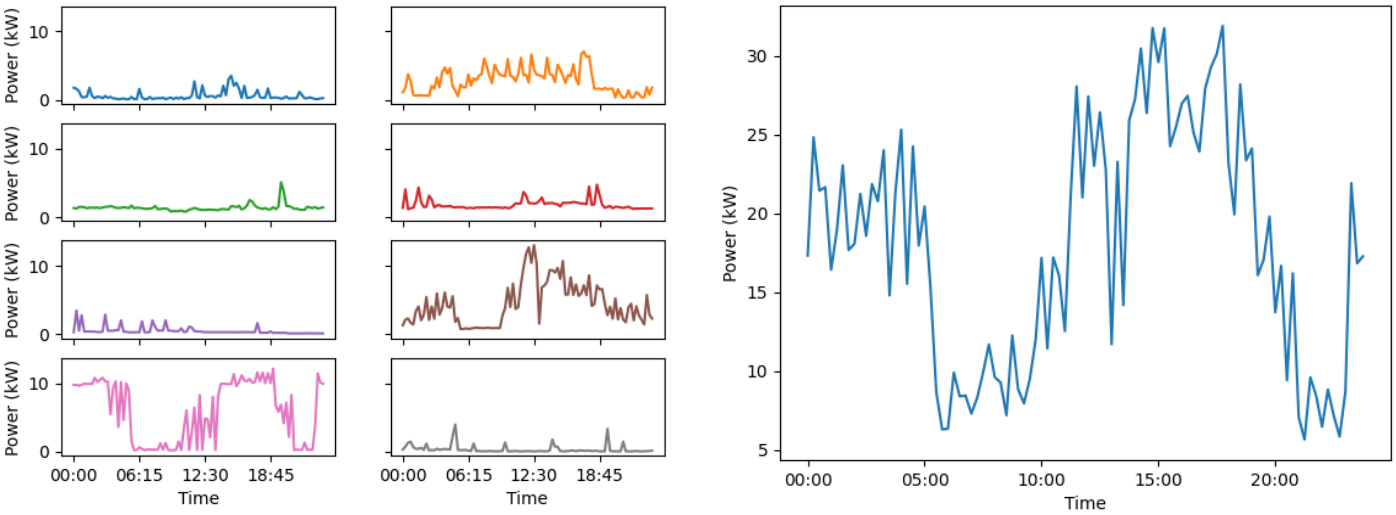

Figure 2: Per-site and total energy consumption readings for building 1 on January 1st, 2023.

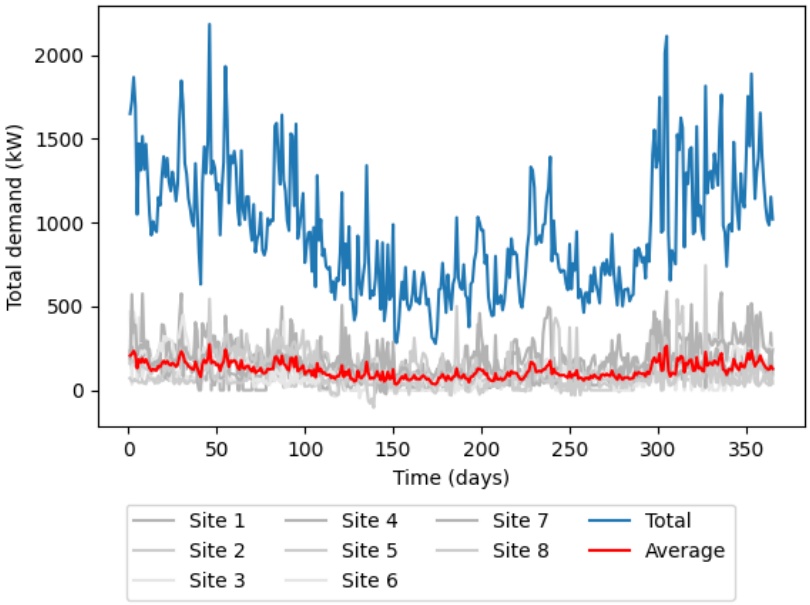

Figure 3: Total, average, and per-site energy demand for building 1 during the year of 2023. Note that the average energy demand for the building could potentially provide a rough estimate of individual site demand patterns if a site is correctly associated with its building.

While the method used for this project can provide some privacy protection for data during processing, it may not be fully adequate to protect data if the resulting statistics are published. Figure 3 shows that sites' energy demands appear to be relatively close to the building's average energy demand. As buildings are created from unaltered data, this similarity could potentially allow a third party to estimate a site's likely energy usage patterns if they know which building the site was placed into during aggregation, though this estimate may not perfectly reflect the actual meter readings. Figure 4 additionally shows that certain features of sites deviating from the average may be preserved in the building's total readings. One site's spike in energy demand between 10:00 AM and 3:00 PM is closely mirrored by the building's total energy demand, and a similar correlation is present for another site between midnight and 5:00 AM. Further measurement of data privacy characteristics is needed to determine the degree of privacy afforded by data aggregation alone.

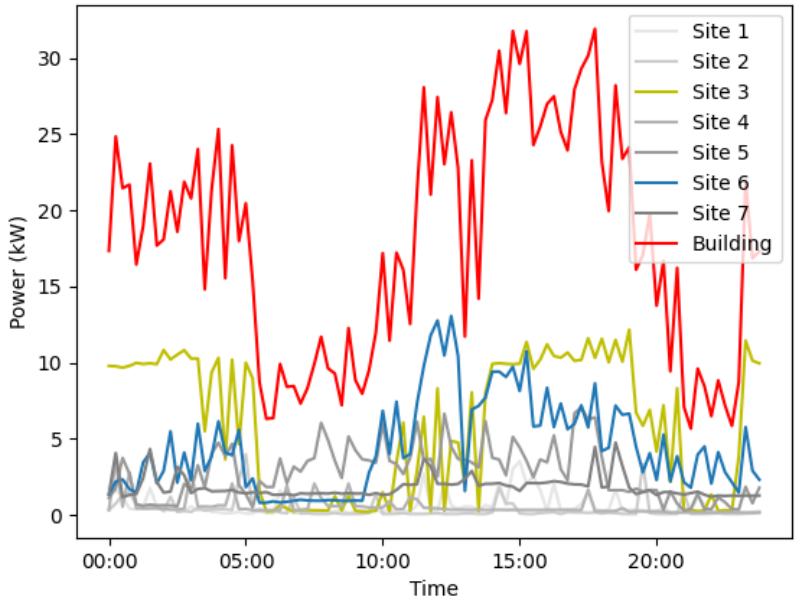

Figure 4: Total vs individual site energy usage for January 1st, 2023. Sites with notable features are highlighted in color. Note how the spike in energy demand for site 6 between 10:00 and 15:00 is visible in the building energy curve, as is the spike for site 3 between 0:00 and 05:00.

Future Work

The Python program created by this project combines data from sites without modifying them. This approach is effective at creating an accurate composite, but it largely does not account for the possibility of an observer having knowledge of which sites are placed in which virtual buildings. To provide better protection of privacy attributes for individual tenants, this approach could be modified to use differential privacy, which intentionally adds noise to data to obscure the original values without significantly distorting the aggregated version of that data. Doing so could make it more difficult to determine an individual site's contribution to the aggregate while preserving the aggregate's usability.

Another area of further work focuses on extending the functionality of this program. At present, it is only able to provide a small selection of statistics. Future work could add other useful statistics that are currently not available; for example, reliability indices could be calculated to provide information on the frequency of power outages for a virtual building. This may require a switch to a fully homomorphic encryption algorithm, as partially homomorphic algorithms do not support comparing values with each other. Many metrics require finding the largest or smallest reading, which is very difficult to do with partially homomorphic encryption without decrypting data. Switching to a fully homomorphic algorithm would allow the computation of these statistics while data is encrypted, but it may negatively impact processing time. Further research needs to be done to determine whether this tradeoff is worthwhile, as well

as into which metrics may be provided without revealing unintended information if those statistics are combined with one another.

At present, statistics can only be verified by calculating them independently with and without encryption. It may be desirable to have a mechanism to validate data accuracy without revealing the original data to the end user. This would assure users that the provided statistics are truthful and have not been altered by the program. Future work could design and implement this verification mechanism as part of this program.

Impact on PNNL Mission

Sustainable energy efforts rely on electrical metering data to determine areas in need of improvement and the success of interventions in those areas. Providing this data directly can infringe on the privacy of building tenants, but previous efforts to provide privacy protections for metering data largely disregarded the need for data availability. This work seeks to supply useful statistics about electrical metering data without revealing information at the level of individual metering sites, which may reassure consumers and encourage further adoption of smart electrical devices and sustainable grid infrastructure.

Conclusions

This project's program aggregates groups of 8 electrical meters into virtual buildings for analysis. While the use of homomorphic encryption to obscure data comes with limitations on speed, this method could offer a degree of privacy protection to building tenants while permitting the calculation of statistics on metering data. Further research is needed to refine this program and investigate additional methods of protecting the privacy of consumer metering data.

References

1. **Hodge, Tyler.** Hourly electricity consumption varies throughout the day and across seasons. *U.S. Energy Information Administration.* [Online] February 2020, 2020. https://www.eia.gov/todayinenergy/detail.php?id=42915.

2. *Toward intelligent demand-side energy management via substation-level flexible load disaggregation.* **Gao, Ang, et al.** August 1, 2024, Applied Energy, Vol. 367.

3. *Novel Temporal Perturbation-Based Privacy-Preserving Mechanism for Smart Meters.* **Wang, Xiaoyan, et al.** October 12, 2019, Mobile Networks and Applications, Vol. 25, pp. 1548-1562.

4. **Livingston, OV, et al.** *Commercial Building Tenant Energy Usage Data Aggregation and Privacy.* 2014. pp. 1-52.

5. **Institute of Electrical and Electronics Engineers.** Types of Homomorphic Encryption. *IEEE Digital Privacy.* [Online] 2024. https://digitalprivacy.ieee.org/publications/topics/types-of-homomorphic-encryption.

6. **Northwest Energy Efficiency Alliance, Inc.** Northwest End Use Load Research Project Energy Metering Data. 2024.

7. *On the Security of Homomorphic Encryption on Approximate Numbers.* **Li, Baiyu and Micciancio, Danielle.** s.l. : Cryptology ePrint Archive, March 7, 2021, Eurocrypt.

8. *Demystifying Bootstrapping in Fully Homomorphic Encryption.* **Badawi, Ahmad Al and Polyakov, Yuriy.** s.l. : Duality Technologies, August 24, 2023, IACR EPrint Cryptology Archive, pp. 1-11.

9. **NIST.** *Recommendation for Key Management: Part 1 - General.* s.l. : National Institute of Standards and Technology , 2020. NIST 800-57.

10. *Selecting Cryptographic Key Sizes.* **Lenstra, Arjen K and Verheul, R Eric.** August 14, 2001, Journal of Cryptology, Vol. 14, pp. 255-293.

11. *Investigating the Effects of varying the Key Size on the Performance of AES Algorithm for Encryption of Data over a Communication Channel.* **Adadeji, Kazeem B and Famoriji, John O.** 8, September 2014, International Journal of Applied Information Systems, Vol. 7, pp. 6-10. ISSN 2249-0868.

12. **Serengil, Sefik Ilkin.** LightPHE: A Lightweight Partially Homomorphic Encryption Library for Python. [Online] 2023. https://github.com/serengil/LightPHE.

13. **Environment Protection Agency.** Building Energy Benchmarking and Transparency: Overview for State and Local Decision Makers. *Energy Star Portfolio Manager.* [Online] February 2021. https://www.epa.gov/system/files/documents/2021-12/section-4-data-access_2-12-2021.pdf.

14. *Comparison of Performance of Data Imputation Methods for Numeric Dataset.* **Jadhav, Anil, Pramod, Dhanya and Krishnan, Ramanathan.** 10, July 4, 2019, Applied Artificial Intelligence, Vol. 33, pp. 913-933.

15. **Gelman, Andrew and Hill, Jennifer.** Missing-data imputation. *Data Analysis Using Regression and Multilevel/Hierarchical Models.* 2007.

Appendix

Acknowledgements

This work was supported and funded in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Community College Internships Program (CCI).

Participants

Name	Institution	Role
David Jonathan Sebastian Cardenas	Pacific Northwest National Laboratory	Mentor. Guided project, reviewed work, answered questions, and provided direction for background research.
Javier E Ramirez	Pacific Northwest National Laboratory	Team member. Answered questions.
Maximillan Yam	Pacific Northwest National Laboratory	Team member. Suggested optimizations for code.
Nancy Roe	Pacific Northwest National Laboratory	Program manager. Provided support for CCI program activities.

Scientific Facilities

No shared user facilities were utilized for this project.

Notable Outcomes

This project will be presented at Pacific Northwest National Laboratory's Research Symposium on August 22nd, 2024.

An abstract for this project was entered into the WDTS Abstract Competition and won the semi-finals at the Pacific Northwest National Laboratory.

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354

1-888-375-PNNL (7665)

www.pnnl.gov