An Open Machine-Learning–Oriented Dynamic Simulation Dataset for Power System Studies

Jiangkai Peng
National Renewable Energy Laboratory
Golden CO, USA
jiangkai.peng@nrel.gov

Ignacio Aravena, Chih-Che Sun
Lawrence Livermore National Laboratory
Livermore CA, USA
aravenasolis1@llnl.gov, sun31@llnl.gov

***Abstract*—Machine learning (ML) methods for power system dynamics studies require access to realistic, diverse, and openly accessible time-series datasets. However, many existing dynamic models are simplified or proprietary, and commercial simulation tools are license-restricted and not designed for large-scale automated batch simulation, limiting the ability to generate dynamic datasets suitable for ML training and evaluation. To address this gap, we develop and release an open, ML-ready dynamic simulation dataset constructed from the 2000-bus Texas synthetic system. The dataset is generated using the open-source ANDES framework and a scalable high-performance computing (HPC) workflow, producing approximately 400,000 time-domain simulations across multiple operating conditions and contingency events. A transient-stability–index based sampling method is introduced to create balanced subsets that preserve diverse dynamic behaviors while significantly reducing dataset size. The dataset, workflow scripts, and documentation are publicly available[1], lowering barriers to applying and evaluating ML methods in power system dynamic studies.**

***Index Terms*—power system, dynamic simulation, machine learning, open-source, dataset.**

I. Introduction

Machine learning (ML) is rapidly emerging as a key enabler for power system studies, advancing applications in transient stability assessment, learning-based control, fault diagnosis, and secure operational planning [1]. These applications depend on access to high-quality dynamic time-series datasets that are realistic, diverse, and openly accessible to support ML model training and evaluation. Large-scale dynamic simulation provides a controlled and fully observable means to generate such data. However, constructing dynamic datasets that meet these requirements remains challenging in practice.

Most existing dynamic simulation datasets used in power system research are limited in accessibility, realism, and scale. Many realistic dynamic models developed by utilities and manufacturers are proprietary due to confidentiality requirements, preventing their release for public research. At the same time, widely used public datasets such as the IEEE benchmark test systems are small in scale and lack detailed control representations of inverter-based resources (IBRs), making them insufficient for studying modern IBR-dominated grids. Furthermore, most commercial dynamic simulation tools are closed-source, require paid licenses, and are not designed for high-performance computing (HPC) workflows such as large-scale parallel execution or automated batch simulation. These factors create practical barriers to generating diverse sets of dynamic simulation results across varied operating conditions and contingency scenarios.

To address these limitations, the research community has increasingly turned to open-source power system models and dynamic simulation tools [2]. Large-scale transmission system models such as the Texas A&M electric grid test cases [3], NREL's reduced Western Electricity Coordinating Council (WECC) models [4], and PNNL's DR POWER repository [5] provide realistic and openly available system representations for stability and planning research. In parallel, open-source dynamic simulation tools including ANDES [6], GridDyn [7], and SIENA [8] offer transparent model structures, user-extensible dynamic component libraries, and support for scalable simulation workflows.

While these models and tools [3]–[8] represent important progress toward openness and realism, they do not, on their own, provide large-scale ML-ready dynamic time-series datasets. Using them to generate such datasets requires (i) integrating a large-scale dynamic model with an open simulation framework, (ii) designing a HPC workflow capable of running tens of thousands of diverse disturbance scenarios, and (iii) curating the resulting time-domain outputs into structured, labeled data suitable for ML training and evaluation. To the best of the authors' knowledge, there is currently no publicly available resource that combines all of these elements into a coherent and openly accessible dataset.

In this work, we release an open, ML–oriented dynamic simulation dataset constructed from the Texas 2000-bus synthetic system (Texas2k), simulated using the ANDES open-source platform and executed through large-scale HPC workflows. The contributions of this work are threefold:

This work was authored in part by NREL for the U.S. Department of Energy (DOE), operated under Contract No. DE-AC36-08GO28308, and performed in part by LLNL under the auspices of the U.S. DOE, under Contract No. DE-AC52-07NA27344. Funding was provided by the U.S. DOE Office of Electricity. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes. Document control numbers: NREL/CP-5D00-98100, LLNL-CONF-2013450.

[1]Dataset available at: https://dataverse.harvard.edu/dataverse/opencoggrid/.

979-8-3315-8138-1

- Conversion and refinement of the Texas2k dynamic models in the ANDES environment, ensuring realistic dynamic response and stable time-domain simulation under contingency conditions.
- Development of a scalable HPC simulation workflow that generates tens of thousands time-domain simulations across diverse system conditions and contingency events, together with a transient-stability–index based sampling method to produce balanced ML-ready subsets.
- Public release of the ML-ready dynamic simulation dataset, workflow scripts, and documentation, lowering barriers to applying and evaluating ML methods in power system studies.

II. Methodology

This section outlines the workflow for simulating the Texas2k synthetic power system in an open-source environment. It integrates a realistic large-scale dynamic power sytem model, an open simulation framework, and a data conversion process that bridges proprietary formats to machine-learning-ready datasets.

A. Texas2k Synthetic System Overview

The Texas2k system, shown in Fig. 1, was developed by Texas A&M University to provide a large-scale, publicly available test case free from confidentiality constraints. It was created through a systematic methodology designed to reproduce the statistical and structural characteristics of real transmission networks while maintaining no physical correspondence to any existing grid. The resulting system features realistic geographic layout, electrical parameters, and dynamic characteristics suitable for steady-state, transient stability, and geomagnetic disturbance studies [3]. The version used in this work reflects 2016, 2024, and 2025 operating conditions, incorporating new transmission facilities, updated generation and load levels, and refined dynamic model tuning to improve transient and small-signal behavior [9].

A summary of the Texas2k system is presented in Table I. The system includes a mix of conventional synchronous generators, PV plants, and wind farms. Each dataset version contains multiple dispatch scenarios, such as summer peak and low-load conditions. The 2024 cases further include high-renewable and low-load configurations with grid-forming inverters to represent emerging IBR-dominated grid conditions.

TABLE I
Texas2k System Dataset Comparasion

Capacity (GW)	SG	PV	Wind	Total
2016 case	103.81	6.73	9.07	119.61
2024 case	103.81	19.57	58.68	182.06
2025 case	102.60	35.90	46.48	184.98

The dynamic model set in the Texas2k system includes a comprehensive range of conventional SG and IBR representations. Conventional SG units are modeled using the GENROU round-rotor synchronous generator model, combined with turbine-governor systems such as TGOV1, IEEEG1, and HYGOV, and excitation systems including EXST1, IEEET1, ESST1A, ESST4B, and IEEEST. IBRs are represented by the WECC-approved second-generation renewable energy models, consisting of the REGCA1 converter model, REECA1 electrical control model, and REPCA1 plant controller, along with WTARA1, WTDTA1, WTPTA1, and WTTQA1 to describe wind turbine aerodynamic, drive-train, pitch, and torque dynamics. The detailed descriptions and parameter definitions of these models can be found in [10], [11], and they follow conventions consistent with commercial power system simulation tools such as PSS®E and PowerWorld Simulator. Together, these models capture the essential control and electromechanical behavior of both SGs and IBRs.

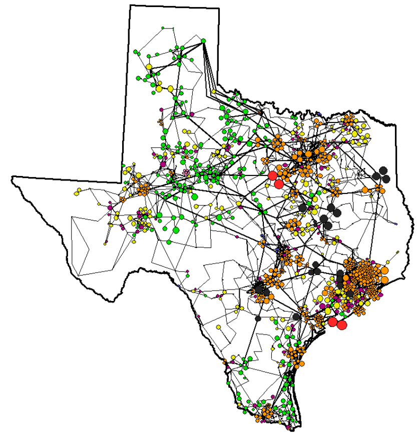


Fig. 1. Texas 2000-bus synthetic power system. [12]

B. ANDES Simulation Framework

ANDES is an open-source Python framework for power system modeling, simulation, and control [6]. It supports power flow, transient stability, and small-signal stability analysis with high numerical efficiency and scalability. The framework includes a comprehensive library of commercial-grade dynamic models, encompassing detailed representations of synchronous machines, exciters, governors, and second-generation renewable energy models. ANDES employs a symbolic–numeric framework for differential–algebraic equation (DAE) formulation, enabling users to describe device equations symbolically and automatically generate optimized numerical code for large-scale computation. Critically for large systems, ANDES supports multi-core and parallel execution, enabling simultaneous simulation of batches or multiple scenarios leveraging modern HPC-architectures. Its results have been validated against commercial software such as TSAT and PSS®E, demonstrating close numerical agreement. These features make ANDES well-suited for large-scale dynamic simulation studies, and we therefore adopt it as the simulation platform in this work.

C. Conversion Workflow

To execute the Texas2k system within the ANDES environment, a comprehensive data conversion process was required.

979-8-3315-8138-1

The Texas2k system was originally developed and validated in PowerWorld Simulator, and its primary distribution is provided in PowerWorld case (.PWB) and auxiliary (.AUX) formats. To facilitate interoperability, the dataset is also available in MATPOWER, PSS®E (.RAW/.DYR), and PSLF (.epc) formats. In this study, the PSS®E RAW and DYR files were used as the primary data source for both steady-state and dynamic model conversion. The process involved data parsing to extract network topology, generator parameters, and dynamic model definitions from the PSS®E files and translate them into the ANDES Excel-based data format (XLSX) required for simulation. Although ANDES includes built-in support for importing PSS®E data, several software issues and inconsistencies were identified and corrected to ensure compatibility. Additionally, some control parameters, particularly within SG exciters and renewable energy models, were found to be physically unrealistic and were adjusted to improve numerical stability and dynamic performance. A summary of the workflow is illustrated in Fig 2.

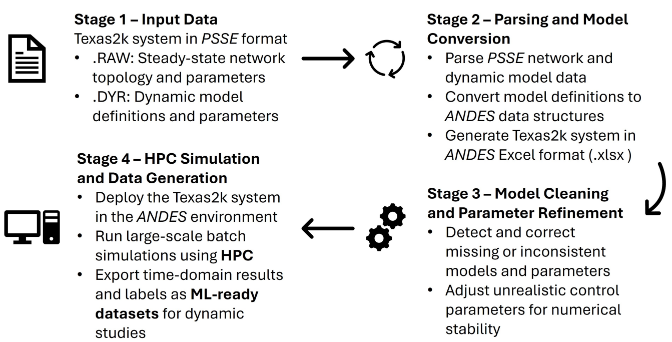


Fig. 2. Data Conversion and HPC Simulation Workflow for the Texas2k System.

III. HPC Simulation Framework

This section describes, first, the HPC workflow we implemented to generate and store our large-scale database of dynamic simulations, and, second, how we sample a balanced subset of the simulations to arrive at a practically-sized dataset.

A. HPC Setup

We run the dynamics simulations using an embarrassingly-parallel `coordinator-worker` workflow (one `coordinator`, many `workers`). The simulation starts from a *system* on a given condition. The `coordinator` holds a list of *events* (*e.g.*, generator outages, line outages, etc.), that it iteratively assigns to the `workers` as they become available. Each `worker` waits for an *event* from the `coordinator`, it applies the *event* to the *system*, run the time domain simulation for the *event*, and save the integration result in a `worker`-specific file. Once all the *events* have been simulated, the `workers` idle and the `coordinator` merges all different `worker` files into a single file for the specified *system* and condition.

This workflow is implemented in Python with MPI communications, specifically, we implement through mpi4py [13], with OpenMPI [14] backend. The time-domain simulation results are extracted from the ANDES system data structures, converted to single-point precision for storage considerations, and saved in HDF5 format using h5py [15].

B. TSI-based Balanced Sampling

Time-domain simulation results are challenging to store because they require maintaining cycle-level time resolution snapshots for thousands of variables across the system. This prevents storing these datasets and transferring them over the internet, thereby preventing the development AI tools that can use them. To address this issue, we downsize the simulation data using a specialized sampling process that looks for events that are representative of the range of behaviors that we observe in our large-scale simulations.

Our sampling process uses Transient Stability Indexes (TSIs) [16] as proxies to discriminate between different types of events. The TSI provides a scalar measure of system-wide rotor-angle coherence during a disturbance. It quantifies the maximum instantaneous spread in electrical rotor angles among all synchronous generators relative to the loss-of-synchronism boundary. Specifically,

$$\mathrm{TSI} = 1 - \frac{\max\limits_{t}\left(\max\limits_{i}\ \delta_i(t) - \min\limits_{i}\ \delta_i(t)\right)}{\pi} \tag{1}$$

where $\delta_i(t)$ is the electrical rotor angle of generator i. A spread of π radians corresponds to the stability limit at which synchronism is lost. Values of TSI closer to 1 indicate tightly clustered generator angles and stable operation, while lower values signal increasing risk of loss of synchrony.

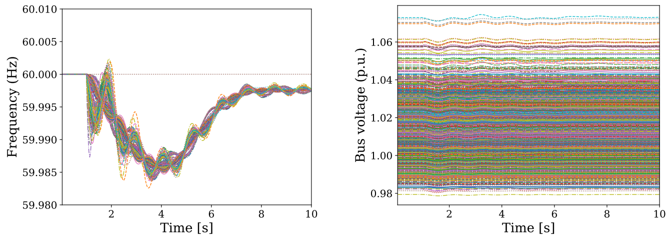
We select a number of TSIs and compute them for every event simulated, thereby, producing a low-dimensional feature vector $\boldsymbol{x}_i$ for every event i and an empirical distribution of those vectors, $\tilde{\mathcal{X}}$. If we were to directly sample from the set of events, with an equal probability associated to each, we would be implicitly sampling from $\tilde{\mathcal{X}}$, which likely presents a very concentrated distribution. This concentration of mass in the distribution of TSIs is a consequence of most events having small effects on the power system stability, a fact often exploited in power systems operations. Sampling from $\tilde{\mathcal{X}}$, therefore, would result in a large proportion of trivial events being sampled for any sample size we chose.

Instead, we are interested in obtaining a sample from the support of $\tilde{\mathcal{X}}$ (i.e., $\boldsymbol{x}_i$ for all i) with an uniform distribution. To achieve this, we proceed as follows:

1) We use multivariate Kernel Density Estimation (KDE) [17] to estimate $\mathcal{X}$, an hypothetical continuous distribution of TSIs, from our TSI data $\tilde{\mathcal{X}}$. Denote this estimate distribution $\hat{\mathcal{X}}$, with density function $f_{\hat{\mathcal{X}}}$.
2) We construct an almost uniform discrete distribution over the support $\hat{\mathcal{X}}$, $\bar{\mathcal{X}}$ with probability mass function

$$p_{\bar{\mathcal{X}}}(\boldsymbol{x}_i) := F / f_{\hat{\mathcal{X}}}(\boldsymbol{x}_i)$$

979-8-3315-8138-1



(a) Frequency (b) Voltage

Fig. 3. Texas2k dynamic simulation results in ANDES.

where $F = 1/(\sum_i 1/f_{\hat{\mathcal{X}}}(\boldsymbol{x}_i))$.

3) We draw our sample from $\bar{\mathcal{X}}$, which will be spread almost uniformly over the support $\tilde{\mathcal{X}}$.

We implement the sampling process above in R, using the KDE functionality within the `ks` package [18] and using the least-squares cross-validation bandwidth matrix selector for multivariate data [19], as it showed more robustness than alternative bandwidth selectors in early testing.

IV. Numerical Results

This section, first, presents a verification of the Texas system converted to ANDES for time-domain simulations, then, it describes our HPC setup and the shows statistics for the simulations carried out, and, finally, it demonstrates our TSI-based approach for balanced sampling.

A. Simulation Verification

Dynamic contingency simulations were performed across different Texas2k cases. The 2016 summer peak case is shown here as a representative example. To validate the converted Texas2k dynamic model in ANDES, an N–1 generator trip was applied in which the 146.4 MW generator at bus 5416 was disconnected at $t = 1$ s. The simulation results are shown in Fig. 3.

Following the generator trip, the system frequency experiences a transient drop of approximately 16 mHz and then recovers as primary frequency control restores power balance. The voltage response exhibits small and well-damped oscillations, and the system returns to a stable post-contingency operating point. These results demonstrate that the converted dynamic model reproduces realistic transient behavior consistent with expected frequency and voltage control dynamics.

B. HPC Simulation Results

We carried out time-domain simulations for the Texas2k system using the embarrasingly-parallel implementation described in §III-A on the Dane cluster [20], hosted at LLNL. Each node of the cluster has dual Intel(R) Xeon(R) Platinum 8480+ with 56 cores at 2.0 GHz processors (112 cores per node) and 256GB of RAM memory. We conduct time-domain simulations for 100,000 events for four cases (system and conditions) derived from the Texas 2k 2016 and 2024 systems. The simulated events correspond to N-1 or N-2, generator and line trip contingency events, with dual events being applied at the $t = 1.0$ seconds and $t = 1.1$ seconds marks, and simulations with $t = 10.0$ seconds horizon and half-cycle step size.

TABLE II
HPC simulation statistics.

Case	# events completed	Failure rate	File size [GB]
2016 Full SG IBR	95,409	4.59%	10,649
2016 Summer Peak	95,806	4.19%	9,507
2024 Summer Peak	96,896	3.10%	6,499
2024 Low Load	97,777	2.22%	4,004

These simulations were conducted in batches of 10,000 events, using allocations of 4 nodes of the Dane cluster, with 2 cores per MPI rank (1 coordinator and 223 workers per batch). These simulations consumed a total of 44,820 core-hours of compute time, averaging 3.34 minutes of wall-clock time per time-domain simulation (including I/O) of a single event. For perspective, note that running these simulations sequentially would take approximately 2.5 years.

Table II presents summary statistics for each of the 4 cases for which we conducted time-domain simulations for events. We observe that the simulations of a fraction of the events for each case were not completed. The great majority of these incomplete simulations (98% of all incomplete runs) correspond to very severe contingencies that drive the system into highly unstable behavior. In such cases, the numerical integration in ANDES may fail to converge at a given time step within the allowed solver iterations, leading to early termination in order to avoid nonphysical results and unnecessary computational cost. The remaining incomplete simulations (2% of all incomplete simulations) were terminated or stalled for unidentified reasons. These incomplete simulations without identified causes, however, represent only 0.1% of the total number of events, which is in line with our expectations for simulations at this scale. Finally, Table II also presents the sizes of the HDF5 files storing the time-domain simulations, which total 29.9 TB of raw data, which is beyond manageable for current off-the-shelf computers.

C. TSI-based Balanced Sampling Results

We reduce the size of the dataset while maintaining the representation of the different phenomena in the raw dataset by drawing samples following the procedure outline in §III-B with two TSIs:

- **Absolute TSI**: measures how close the system comes to loss of synchronism based on the maximum rotor-angle separation. Lower values indicate more severe angle divergence.
- **Excursion TSI**: measures how strongly the disturbance excites rotor-angle swings relative to pre-fault values. Lower values indicate larger transient excursions.

Figure 4 shows that, As expected, the empirical joint distribution of Absolute TSI and Excursion TSI is very concentrated. Directly sampling the data, therefore, would result in significant information loss in the sampled dataset, with

979-8-3315-8138-1

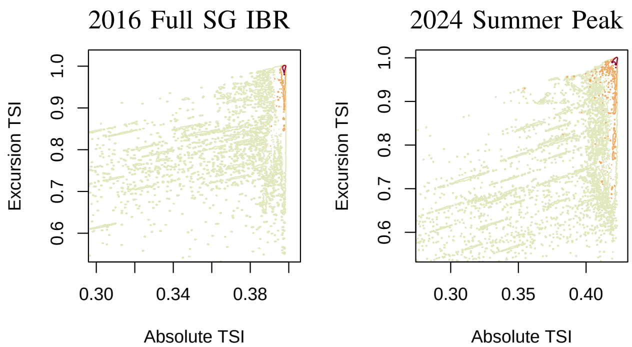


Fig. 4. Estimated density (KDE, $\hat{\mathcal{X}}$) of Absolute TSI and Excursion TSI for two representative cases. Density is presented as contour plots, with red encircling 50% of the probability mass, orange encircling 75% of the probability mass, and olive encircling 99% of the probability mass.

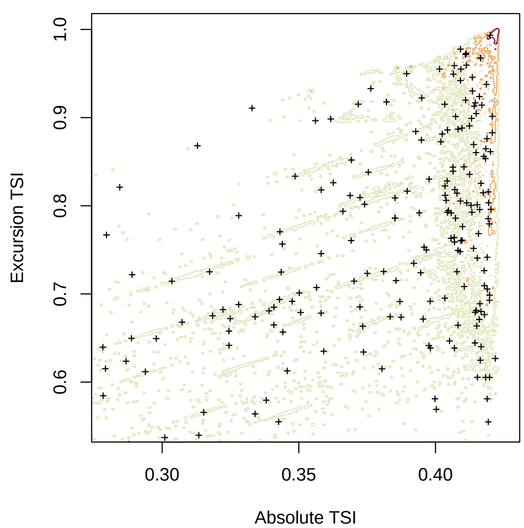


Fig. 5. Balanced sample of 250 events overlayed on the estimated density for the complete dataset for the 2024 Summer Peak case.

most sampled points contained within a very small region of the range of behaviors.

Instead, following our balanced sampling procedure, we obtain samples that span the entire feature space, as shown in Figure 5 for the Texas2k 2024 Summer Peak case. The ability to obtain small informative samples allows us to produce significantly smaller datasets, *e.g.*, reducing the dataset from 29.9 TB to 80 GB when producing samples of 250 events per case.

V. Conclusions

We presented an open, ML-ready dynamic simulation dataset for power system studies, constructed from the 2000-bus Texas synthetic system and refined in the ANDES environment for realistic and stable dynamic behavior. A scalable HPC workflow was developed to generate tens of thousands contingency-based time-domain simulations across multiple operating conditions. A TSI-based sampling method was then applied to create balanced subsets suitable for ML training and evaluation. The dataset and workflow are publicly released to lower barriers to applying and evaluating ML methods in power system dynamic studies.

References

[1] T. Su, J. Peng, A. Selim, J. Zhao, and J. Tan, "A survey of open-source power system dynamic simulators with grid-forming inverter for machine learning applications," in *2025 IEEE Power & Energy Society General Meeting (PESGM)*, 2025, pp. 1–5.

[2] I. Aravena, C.-C. Sun, R. Shi, S. Majumder, W. Yan, J.-Y. Joo, L. Xie, and J. Wang, "Open power system datasets and open simulation engines: A survey toward machine learning applications," *IEEE Open Access Journal of Power and Energy*, vol. 12, pp. 353–365, 2025.

[3] S. Kunkolienkar, F. Safdarian, J. Snodgrass, A. Birchfield, and T. Overbye, "A description of the Texas A&M University electric grid test case repository for power system studies," in *2024 IEEE Texas Power and Energy Conference (TPEC)*, 2024, pp. 1–6.

[4] H. Yuan, R. S. Biswas, J. Tan, and Y. Zhang, "Developing a reduced 240-bus WECC dynamic model for frequency response study of high renewable integration," in *2020 IEEE/PES Transmission and Distribution Conference and Exposition (T&D)*, 2020, pp. 1–5.

[5] M. J. Rice, S. T. Elbert, O. A. Kuchar, D. Pinney, and L. D. Marinovici, "Data repository for power system open models with evolving resources (DR POWER) (final scientific/technical report)," Pacific Northwest National Laboratory (PNNL), Richland, WA (United States), Tech. Rep., 09 2020. [Online]. Available: https://www.osti.gov/biblio/1761209

[6] H. Cui, F. Li, and K. Tomsovic, "Hybrid symbolic-numeric framework for power system modeling and analysis," *IEEE Transactions on Power Systems*, vol. 36, no. 2, pp. 1373–1384, 2021.

[7] Lawrence Livermore National Laboratory, "Griddyn: an open-source power transmission simulation software package," 2025, accessed: 2025-11-10. [Online]. Available: https://github.com/LLNL/GridDyn

[8] National Renewable Energy Laboratory (NREL) – Sienna Team, "Sienna: Open-source energy systems modeling platform," 2025, accessed: 2025-11-10. [Online]. Available: https://github.com/NREL-Sienna

[9] J. Baek and A. B. Birchfield, "A tuning method for exciters and governors in realistic synthetic grids with dynamics," in *2023 North American Power Symposium (NAPS)*, 2023, pp. 1–6.

[10] "WECC approved dynamic model library," Western Electricity Coordinating Council (WECC), Tech. Rep., May 2025. [Online]. Available: https://www.wecc.org/sites/default/files/documents/progress_report/2025/Approved%20Dynamic%20Models%20May%202025.pdf

[11] "Model user guide for generic renewable energy system models," Electric Power Research Institute (EPRI), Tech. Rep. 3002014083, July 2018. [Online]. Available: https://restservice.epri.com/publicdownload/000000003002014083/0/Product

[12] "Texas A&M University electric grid datasets," 2025, accessed: 2025-10-27. [Online]. Available: https://electricgrids.engr.tamu.edu/texas2k-series25/

[13] L. Dalcin and Y.-L. L. Fang, "mpi4py: Status update after 12 years of development," *Computing in Science & Engineering*, vol. 23, no. 4, pp. 47–54, 2021.

[14] E. Gabriel, G. E. Fagg, G. Bosilca, T. Angskun, J. J. Dongarra, J. M. Squyres, V. Sahay, P. Kambadur, B. Barrett, A. Lumsdaine, R. H. Castain, D. J. Daniel, R. L. Graham, and T. S. Woodall, "Open MPI: Goals, concept, and design of a next generation MPI implementation," in *Proceedings, 11th European PVM/MPI Users' Group Meeting*, Budapest, Hungary, September 2004, pp. 97–104.

[15] A. Collette, *Python and HDF5*. O'Reilly Media, Nov. 2013.

[16] L. Shi, S. Dai, Y. Ni, L. Yao, and M. Bazargan, "Transient stability of power systems with high penetration of dfig based wind farms," in *2009 IEEE Power & Energy Society General Meeting*, 2009, pp. 1–6.

[17] J. E. Chacón and T. Duong, *Multivariate kernel smoothing and its applications*. London, England: CRC Press, Jun. 2020.

[18] T. Duong, "ks: Kernel density estimation and kernel discriminant analysis for multivariate data in r," *Journal of Statistical Software*, vol. 21, no. 7, p. 1–16, 2007.

[19] C. J. Stone, "An asymptotically optimal window selection rule for kernel density estimates," *The Annals of Statistics*, vol. 12, no. 4, pp. 1285–1297, 1984. [Online]. Available: http://www.jstor.org/stable/2241002

[20] Lawrence Livermore National Laboratory, "Dane — HPC LLNL," 2025, accessed: 2025-11-07. [Online]. Available: https://hpc.llnl.gov/hardware/compute-platforms/dane

979-8-3315-8138-1