Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2024 SCPE. Volume 25, Issues 5, pp. 3841–3849, DOI 10.12694/scpe.v25i5.3022

IMPLEMENTATION AND OPTIMIZATION OF PROBABILISTIC AND MATHEMATICAL STATISTICAL ALGORITHMS UNDER DISTRIBUTIVE ARCHITECTURE

SHENGBIAO LI*AND JIANKUI PENG†

Abstract. Statistical methods must be developed and optimized in distributed systems due to the increasing amount of data and processing demands in modern applications. The application and optimization of mathematical and probabilistic statistical methods in distributed computing settings is the main topic of this study. Algorithms like these have the potential to improve performance, scalability, and parallel processing abilities when integrated into distributed systems. We commence our investigation by reviewing current mathematical and probabilistic statistical algorithms, determining their advantages and disadvantages, and evaluating their suitability for distributed architectures. We then suggest new approaches for their smooth incorporation into distributed computing structures, making use of distributed storage and parallel processing to effectively manage massive datasets. Improving these algorithms' performance in distributed environments is the focus of this research's refinement phase. We seek to optimize the use of distributed infrastructures by minimizing latency and maximizing computational resources by investigating efficient communication protocols, load balancing mechanisms, and parallelization approaches. The suggested algorithms are put into practice inside a distributed structure for empirical confirmation, and their effectiveness is evaluated in comparison to more conventional, non-distributed competitors. We test the scaling, precision, and effectiveness of the methods in practical scenarios using a variety of datasets and use cases.

Key words: probabilistic optimization; stochastic optimization; robust optimization; distributional robust optimization; chance constrained optimization; energy management; smart grid

1. Introduction.

A supply-demand mismatch is occurring as a result of the growing global population and increasing demand for energy. Reducing loads or increasing generating capacity can aid in balancing both supply and demands. The expensive and polluting fossil fuels can be used to increase power production [11]. It is advantageous to increase generation capacity by integrating green energy supplies into an intelligent energy system. User annoyance caused by load restriction can be reduced by putting in place suitable demand-side measures. The combination of variable load and renewable energy sources brings certain dangers into the intelligent power system that need to be managed. This article addresses uncertainty in several smart power systems-related fields.

Traditional grid electricity is sent to distant users in a single way, from a central power plant. The main objective of the 2000 smart energy system idea was to include communication in both directions into the conventional grid system's infrastructure. A smart power system connects the power plant to the customers. technology of information and communication [7, 16]. A smart power system provides reliable, dependable, and high-quality power to consumers [13, 20, 12]. Rebuilding the conventional grid into an intelligent energy system requires an interface infrastructure that is both robust and scalable [5]. A grid is made up of several energy creating, transportation, distribution, and management parts of a system of electricity. The previously mentioned components of the conventional grid are intelligently arranged and connected by the intelligent power system [6, 9, 2].

The main component of a smart power system is a producing station. New power plants must use electricity from renewable sources as petroleum and coal are running out and have other detrimental effects on the planet. Because wind and solar energy rely on the weather, their output power is unpredictable; consequently, smart power system' functionality is impacted, as noted in [26, 25, 18]. Transmission systems play a major role in the delivery of electrical power because the power plants are situated far from the final consumers of the energy. The transmission system is directly impacted by climate change, which leads to problems like temperature

*School of Education ,Lanzhou University of Arts and Science, Lanzhou,73000, China (shengbiaolire@outlook.com)
†School of Education ,Lanzhou University of Arts and Science, Lanzhou,73000, China

and wind stress. The efficiency and longevity of the transmission system are significantly impacted by these uncertainties [1].

The main contribution of the proposed method is given below:

1. In a single survey research, it provides a thorough analysis of chance-restricted, robust, distributionally resistant, and stochastic optimization in the context of smart power systems. Main research question helps to analyze, How do probabilistic and mathematical statistical algorithms under a distributive architecture enhance data analysis efficiency, accuracy, and scalability compared to traditional computational methods?
2. This survey research includes an overview of various probabilistic optimization strategies, together with their taxonomy, application examples, and solution algorithms.
3. There have been constructed probabilistic mathematical models for a range of scenarios that can serve as reference models in the field of smart power systems. CNN-LSTM is utilized in smart grid optimization.

Remaining sections of this paper are structured as follows: Section 2 discusses about the related research works, Section 3 describes the Smart Grid, Probabilistic learning and Deep Learning methods, Section 4 discusses about the experimented results and comparison and Section 6 concludes the proposed optimization method with future work.

2. Related Works. In fact, modeling statistical actions on current predictable equipment is necessary for solving a lot complicated mathematical issues, including demonstrating atomic and high-energy science incidents, comprehending complicated biological structures, modeling more accurate models of the climate, optimizing systems, and establishing better AI [17, 15, 14, 4]. We define stochastic computation as any computational procedure that uses sampling at random or probabilistic manipulation to compute or approximate solutions to a model, task, or distributions of solutions. Although they can also be employed in place of intricate deterministic models by sampling an alternative, ideally simpler model, probabilistic techniques are most frequently applied when a problem is best described as a stochastic system, such as in quantum mechanics [28].

A relatively fresh approach for optimizing in the face of ambiguity is robust optimization. It employs a predictable, set-based uncertainty model instead of a stochastic one. Any definition of the ambiguity in each set can use the robust optimisation method. Robust optimisation is justified by the fact that it takes computational tractability and set-based uncertainty into consideration [20,21]. Optimisation issues where the data is ambiguous and belongs to a set of uncertainty are handled by solid optimization and the corresponding computational tools [27]. Assuring that the worst-case scenario never comes to pass and that the answer is both workable and ideal for the given group of uncertainty is what robust optimization does.

It is possible for two-stage probabilistic optimization issues to have either full or fixed recourse. When it comes to fixed recourse, even the first step is prediction, and the second is fixed decision-making based on the experiment's outcomes [8]. Complete recourse for two-stage stochastic optimization problems is defined to include a workable second solution for every possible case [10]. Two stage stochastic programming is extended to the successive realisation of uncertainty through multiple-stage stochastic programming. Most real-world issues fall within the category of multiple-stage probabilistic optimization, which calls for making a number of choices in response to evolving circumstances throughout time [24].

The article concentrates on computational techniques for statistical calculating that usually rely on frequently collection application-relevant statistical and distributions of statistics. Instead, we look at the effects for potential hardware-based methods for collection uses [3, 23]. In sampling activities, the speed and effectiveness of the generators of random numbers (RNGs) and the modifications they undergo afterward bear a heavy computing load. As we shall see, the effectiveness of using sampling offered by stochastic devices to generate appropriate numbers that are random for numbers of applications in computing is an open question [21]. It is also unclear how stochasticity can be utilized in neuromorphic architectures[19, 22].

3. Proposed Methodology. Robust minimization has several applications with dynamic objectives in a smart power system. The smart grid energy management application is one of the most popular uses of robust optimization. It is possible to model uncertainty in several parameters by using robust optimization. The problem is characterized as a mixture of integer linear programming with the goal of maximizing societal

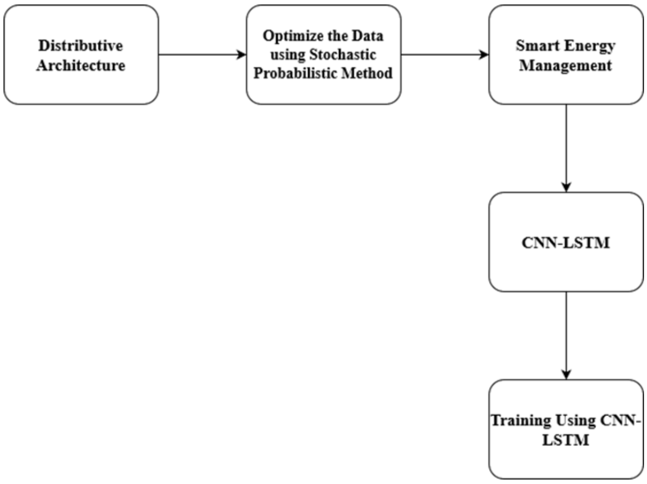

Fig. 3.1: Architecture of Proposed Method

welfare. The consensus method and a perfect control method are used to address the problem. In figure 3.1 shows the Architecture of Proposed Method.

Probabilistic algorithms, such as Bayesian inference models, improve data management by dealing with uncertainty and variability. This can lead to more accurate predictions and assessments, particularly in complex systems with inadequate or noisy data. Distributive systems, such as those used in cloud computing and parallel processing frameworks, allow for the handling of massive datasets. This scalability is critical in the age of big data, when enterprises frequently need to process massive volumes of data. Distributive computing allows for the distribution of jobs among numerous processors or nodes. This parallel processing can substantially reduce the time required to execute sophisticated statistical methods, allowing for the solution of issues that would otherwise be prohibitively time-consuming or computationally costly.

3.1. Microgrid Energy Management. Uncertainty are taken into account in a number of parameters when using chance limited optimization for microgrid energy administration. Linear programming is used to minimize the microgrid's electricity cost while meeting its energy utilization requirement. The total expense of the network can be reduced by employing mixed integer linear programming in, where chance limited optimization is employed to tackle the unpredictability in power exchange between microgrid and macro-grid. Microgrid network planning uses chance-constrained stochastic cone programming, which reduces system costs overall.

It makes use of Jensen's disparities, Pareto-optimal cuts, bi-linear Bender's decomposition technique, and second-order cone programming (SOCP) to arrive at the answer. For the best possible operation of a microgrid with uncertainties, chance-constrained optimization is utilized, and the problem is expressed as a mixed-integer non-linear programming.

3.2. Distributed Energy Management. Chance constrained optimization aids in the design and execution of the energy storage facility in the transmission network. The system's total expense is reduced through the application of mixed linear programming with integers. Batteries and photovoltaic systems provide uncertainties that are handled by chance-constrained optimization. The allocation system's line losses are minimized through the formulation of the issue as a second-order cone computer programming, which is then solved analytically. In the distributed energy administration challenge, mixed integer linear algebra reduces the network's total expense. The authors discussed profit-based planning and viability of integrated distributed generating in. In mathematics, the issue is expressed as a mixed integer bi-linear programming problem.

3.3. Unit Commitment. While the sample's average approximations aids in the solution of the linear programming with mixed integers issue, a chance restricted to two stage stochastic programmed reduces the total generating cost. Possibility limited optimization is used to optimize spinning reserve cost under an uncertain controllable load. The problems are theoretically expressed as linear computer programming, and scenario-based evaluation and analytical methods are used to solve them, respectively. Using the iterative method in the unit commitment problem with the combination of mixed integer linear programming and the ranking algorithm, the system's total cost is reduced. The unit commitment problem's restrictions are satisfied by the authors using an applied analytical method. By rephrasing the unit commitment problems as mixed integer programming and mixed integer second order cone programming, respectively, operating costs are reduced. Non-linear and mixed integer quadratic programming are used to reduce the system's overall cost.

3.4. Optimization using LSTM-CNN. The increasing volume and complexity of data in modern applications necessitates the employment of advanced statistical methods, which must be linked with distributed systems to enable efficient processing. This research focuses on the use of Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs) to enhance probabilistic and mathematical statistical approaches in a distributed setting. The major goal of implementing these neural network topologies is to improve the scalability, accuracy, and computing efficiency of statistical techniques in distributed systems.

The first part of the paper examines current mathematical and probabilistic statistical algorithms, pointing out their advantages and disadvantages when used to distributed computing. Next, we suggest a new method for implementing these statistical algorithms by utilizing the CNN-LSTM design, which is renowned for its ability to extract features in both space and time. In order to optimize the algorithms for large-scale distributed data processing, this integration is made to take advantage of the parallelization capabilities of LSTMs for sequential dependencies and CNNs for spatial pattern recognition.

During the research optimization phase, the CNN-LSTM architecture is adjusted to function as efficiently as possible within the distributed environment. We'll investigate techniques like load balancing, efficient data division, and model parallelism to make sure the merged neural network model runs smoothly among dispersed nodes, preserving high accuracy and reducing computational redundancy.

Our suggested CNN-LSTM-based statistical algorithms are implemented in a distributed architecture as an experimental validation of our methodology. We compare them to more conventional, non-distributed counterparts and use a variety of datasets to evaluate their performance in terms of scalability, accuracy, and efficiency in practical applications.

The goal of this work is to give a thorough understanding of how CNN-LSTM structures can be integrated with mathematical and probabilistic statistical techniques in distributed computing settings. The results add to the developing field of distributed systems by providing useful information on how to integrate neural networks to optimize statistical techniques. Furthermore, the outcomes lay the groundwork for future developments in the fields of statistical computing, distributed architectures, and machine learning.

Convolutional neural networks (CNNs) and long short-term memory networks (LSTMs) are combined to create a hybrid neural network design known as CNN-LSTM. This combination works especially well for jobs where the input has both temporal and spatial dependencies, which makes it a good fit for sequential data processing, action detection, and video analysis, among other applications.

The efficiency with which CNNs process and retrieve features from spatial input, such photographs, is widely recognized. Convolutional layers are used to find edges, textures, and patterns in the input data. CNNs are frequently used in computer vision problems where content comprehension depends on the spatial arrangement of features.

Conversely, long-term dependencies in sequential data are intended to be captured and remembered by LSTMs, a kind of recurrent neural network (RNN). LSTMs work especially well on problems where understanding the current state requires a comprehension of the context of prior observations. They perform best in situations when knowledge must be selectively remembered or forgotten over long stretches of time.

The CNN layers constitute the CNN-LSTM architecture's initial level. Their main job is to process spatial data. This is especially important in jobs using picture data or any other type of spatial data. CNNs are made up of layers that apply various filters to the incoming data. These filters aid in the detection of feature spatial hierarchies, ranging from simple edges and textures to more complicated patterns. Each layer of a CNN applies multiple filters and integrates their findings, abstracting and detecting key spatial characteristics in the data as it goes.

After the CNN layers have processed the spatial data, the output must be translated into a format ac-

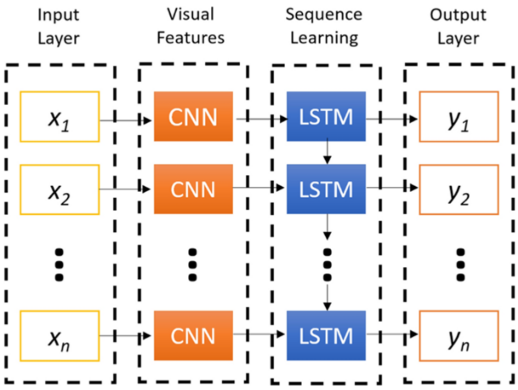

Fig. 3.2: Structure of CNN-LSTM

ceptable for the LSTM layers. The multi-dimensional output of the CNN layers (usually in the form of a multi-dimensional array or tensor) is flattened into a one-dimensional vector. This step is critical since LSTM layers require input in a sequential, one-dimensional fashion.

The LSTM layers comprise the CNN-LSTM architecture's second step. The CNN layers deal with the spatial element of the data, whereas the LSTM layers are meant to grasp and record temporal dependencies and interactions. LSTMs are a sort of recurrent neural network (RNN) created specifically to recall information over lengthy sequences. Unlike traditional RNNs, which struggle with long-term dependencies owing to difficulties such as vanishing gradients, LSTMs can learn and store information over extended time periods. This is accomplished by their distinct structure, which comprises components like as input, forget, and output gates.

CNN Layers. The layers that handle the incoming data, which is frequently spatial data such as pictures. Important spatial trends and features are extracted by the CNN layers.

Flattening. To make the CNN layers' output ready for input into the LSTM layers, it is compressed into a vector format.

LSTM Layers. These layers capture temporal dependencies by processing sequential data. Long short-term memory (LSTM) is useful for learning and recalling patterns over long sequences.

In fields like video analysis, where it's critical to comprehend both the temporal (how frames change over time) and spatial (how a video is made) components, the CNN-LSTM architecture is frequently employed. Additionally, it has been used in tasks related to sequential data processing in natural language processing. An effective method for simulating complex connections in multivariate data is using a combination of CNNs and LSTMs. Figure 3.2 shows the structure of CNN-LSTM.

3.5. Demand Side Management. Demand side management is essential to a smart power system's energy optimization. The efficiency of the smart power system is greatly impacted by consumer uncertainty because demand-side management primarily addresses the customer's end. Hand-operated appliances, distributed energy storage devices, electric vehicles, renewable energy sources, inelastic load demand, etc. are some of the components that create uncertainty for customers. Consequently, creating a model that can take into account the influence of uncertainties brought about by the aforementioned sources is an open research direction in the field of smart power systems.

3.6. Integration of Distribution Energy Resources. Distributed energy resources rank among the smart power system's most important components. Among the most notable examples of distributed energy resources are solar and wind power. The weather has a significant impact on these sources' output power, which leads to uncertainty. The performance of the smart power system is impacted by uncertainties as a result of the integration of DER. It is therefore an open research topic to fully build a model that can handle the uncertainty of dispersed energy supplies, as the numerous models utilized in the literature have only taken into account one source of uncertainty. Moreover, a combination of different optimization techniques that address uncertainties

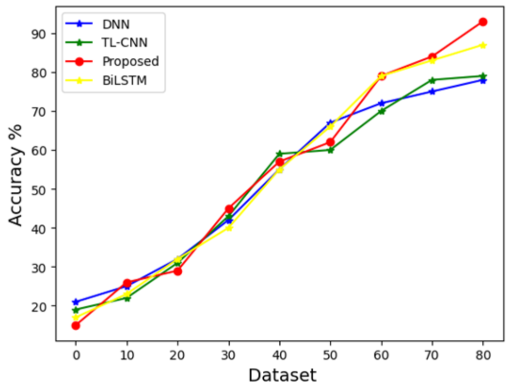

Fig. 4.1: Accuracy

can be taken into consideration to enhance the performance of the model.

4. Result Analysis. The study's findings, involving ACC, F1, Kappa, and each test group's prediction time as determined by the suggested CNN-LSTM deep learning techniques, are shown in this subsection.

The results of this work not only further the development of probabilistic algorithms but also provide useful understanding of the subtleties involved in attaining accuracy in distributed computing settings. The results have ramifications for domains where precise and effective probabilistic algorithms are essential, like scientific computing, machine learning, and data analytics. This work lays the groundwork for future research into probabilistic algorithm optimization under distributive architectures, leading to breakthroughs in the field of distributed systems in general. In figure 4.1 shows the evaluation of Accuracy.

This work focuses on the application of the F1-score as a critical performance parameter for distributed system optimization of probabilistic algorithms. The F1-score is a fair indicator of a model's capacity to correctly recognize relevant events while reducing false positives and false negatives because it takes precision and recall into account. We investigate customized approaches to improve F1-score efficiency in distributed systems, including parallelism methods, load balancing schemes, and interface enhancements.

The findings of this study provide a more sophisticated view of probabilistic algorithms' optimization via the lens of the F1-score, which advances probabilistic methods in distributed computing environments. Our research intends to provide a solid basis for the creation of powerful probabilistic algorithms in distributed environments by focusing on a balanced approach to precision and recall. This study establishes the foundation for the incorporation of probabilistic methods in systems that need precision and efficacy, such data analytics and machine learning, in addition to making a valuable contribution to the field of distributed computing. In figure 4.2 shows the evaluation of F1-score.

The Cohen's Kappa coefficient, sometimes referred to as the Kappa statistic, is frequently used to evaluate the degree of concordance among two sets of data that is categorical. Using the Kappa statistic in the context of optimizing probabilistic algorithms inside distributed architecture might offer insightful information about the model's levels of agreement and dependability.

Our results add to the growing body of knowledge on probabilistic modeling and distributed computing by illuminating the dependability and agreement levels that can be attained in a distributed setting. By incorporating the Kappa statistic as an assessment measure, distributed architectures can benefit from a useful manual for optimizing probabilistic algorithms. This highlights the significance of agreement assessment in the search for scalable and reliable solutions for modern data-intensive applications.

Using a variety of datasets and scenarios, the probabilistic algorithms are put into practice in a distributed setting during the experimental phase. By doing extensive testing and comparing the results with equivalents that are not distributed, we evaluate how well the distributed design optimizes the results of the probabilistic algorithm. In figure 4.3 shows the evaluation of Kappa Value.

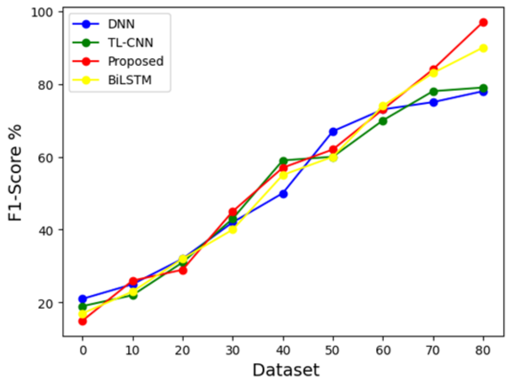

Fig. 4.2: F1-Score

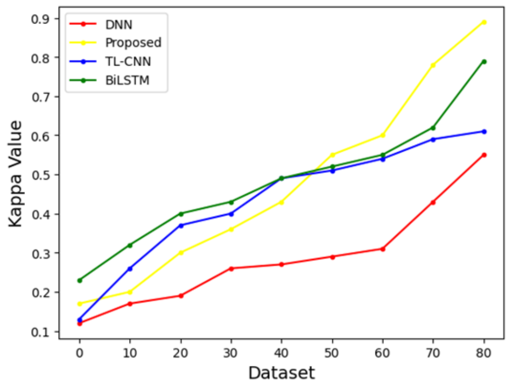

Fig. 4.3: Kappa Value

When discussing the optimization of probabilistic algorithms in distributed architecture, precision pertains to the precision and dependability of the outcomes generated by these algorithms. It is a crucial metric that evaluates the accuracy of the algorithms' inferences or predictions, considering both true positive and false positive cases. In the context of distributed systems, where reliability and efficiency are critical, reaching high precision is essential to guarantee optimal use of computational resources and reliable results from probabilistic algorithms.

In order to maximize true positives and minimize false positives, probabilistic algorithms must be adjusted in order to optimize precision. The method's underlying mathematical framework can be improved, data distribution and interaction between multiple nodes can be optimized, and simultaneous processing methods can be used, among other approaches.

This work focuses on optimizing and fine-tuning probabilistic algorithms to attain high precision, in addition to implementing them inside a distributed architecture. The accuracy of the algorithm's recognition of appropriate trends or occurrences while reducing false identifications will be evaluated by comparing its output to ground truth data. In figure 4.4 shows the evaluation of Precision.

5. Conclusion. Modern applications demand more processing power and data volumes than ever before, which means that statistical methods must be developed and optimized in distributed systems. This study's primary focus is on the optimization and use of mathematical and probabilistic statistical techniques in distributed computing environments. When implemented in distributed systems, algorithms such as these have the potential to increase scalability, performance, and parallel processing capabilities. We first evaluate

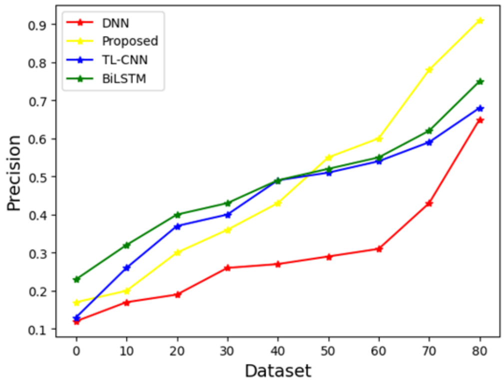

Fig. 4.4: Precision

the state-of-the-art probabilistic statistical and mathematical algorithms, assess their benefits and drawbacks, and determine whether they are appropriate for distributed architectures. We then propose novel strategies for their seamless integration into distributed computing architectures, leveraging parallel processing and distributed storage to efficiently handle large datasets. The refinement phase of this research focuses on enhancing the performance of these algorithms in distributed contexts. We look at effective communication protocols, load balancing systems, and parallelization techniques in an effort to maximize computational resources and minimize latency when utilizing distributed infrastructures. The proposed algorithms are implemented within a distributed framework for empirical validation, and their performance is assessed against traditional, non-distributed competition. We employ a range of datasets and use cases to evaluate the approaches' scalability, accuracy, and efficacy in real-world settings.

Acknowledgment. 2022 Gansu Province higher education innovation fund project under Grant No. 2022A-174.

REFERENCES

[1] J. B. Aimone, R. Lehoucq, W. Severa, and J. D. Smith, *Assessing a neuromorphic platform for use in scientific stochastic sampling*, in 2021 International Conference on Rebooting Computing (ICRC), IEEE, 2021, pp. 64–73.
[2] W. A. Borders, A. Z. Pervaiz, S. Fukami, K. Y. Camsari, H. Ohno, and S. Datta, *Integer factorization using stochastic magnetic tunnel junctions*, Nature, 573 (2019), pp. 390–393.
[3] X. Cao, J. Wang, and B. Zeng, *Networked microgrids planning through chance constrained stochastic conic programming*, IEEE Transactions on Smart Grid, 10 (2019), pp. 6619–6628.
[4] Y. Chen, Q. Guo, H. Sun, Z. Li, W. Wu, and Z. Li, *A distributionally robust optimization model for unit commitment based on kullback–leibler divergence*, IEEE Transactions on Power Systems, 33 (2018), pp. 5147–5160.
[5] V. Demin, I. Surazhevsky, A. Emelyanov, P. Kashkarov, and M. Kovalchuk, *Sneak, discharge, and leakage current issues in a high-dimensional 1t1m memristive crossbar*, Journal of Computational Electronics, 19 (2020), pp. 565–575.
[6] E. J. Fuller, S. T. Keene, A. Melianas, Z. Wang, S. Agarwal, Y. Li, Y. Tuchman, C. D. James, M. J. Marinella, J. J. Yang, et al., *Parallel programming of an ionic floating-gate memory array for scalable neuromorphic computing*, Science, 364 (2019), pp. 570–574.
[7] I. Hussain, G. Samara, I. Ullah, and N. Khan, *Encryption for end-user privacy: A cyber-secure smart energy management system*, in 2021 22nd International Arab Conference on Information Technology (ACIT), IEEE, 2021, pp. 1–6.
[8] R. A. Jabr, *Distributionally robust cvar constraints for power flow optimization*, IEEE Transactions on Power Systems, 35 (2020), pp. 3764–3773.
[9] J. Kaiser, W. A. Borders, K. Y. Camsari, S. Fukami, H. Ohno, and S. Datta, *Hardware-aware in situ learning based on stochastic magnetic tunnel junctions*, Physical Review Applied, 17 (2022), p. 014016.
[10] J. Liu, H. Chen, W. Zhang, B. Yurkovich, and G. Rizzoni, *Energy management problems under uncertainties for grid-connected microgrids: A chance constrained programming approach*, IEEE Transactions on Smart Grid, 8 (2016), pp. 2585–2596.
[11] S. A. Malik, T. M. Gondal, S. Ahmad, M. Adil, and R. Qureshi, *Towards optimization approaches in smart grid a review*, in 2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), IEEE, 2019,

pp. 1–5.

[12] M. J. Marinella, S. Agarwal, A. Hsia, I. Richter, R. Jacobs-Gedrim, J. Niroula, S. J. Plimpton, E. Ipek, and C. D. James, *Multiscale co-design analysis of energy, latency, area, and accuracy of a reram analog neural training accelerator*, IEEE Journal on Emerging and Selected Topics in Circuits and Systems, 8 (2018), pp. 86–101.

[13] M. I. Mosaad, A. Abu-Siada, M. M. Ismaiel, H. Albalawi, and A. Fahmy, *Enhancing the fault ride-through capability of a dfig-wecs using a high-temperature superconducting coil*, Energies, 14 (2021), p. 6319.

[14] S. Prabakaran, R. Ramar, I. Hussain, B. P. Kavin, S. S. Alshamrani, A. S. AlGhamdi, and A. Alshehri, *Predicting attack pattern via machine learning by exploiting stateful firewall as virtual network function in an sdn network*, Sensors, 22 (2022), p. 709.

[15] Y. Sasaki, N. Yorino, Y. Zoka, and F. I. Wahyudi, *Robust stochastic dynamic load dispatch against uncertainties*, IEEE Transactions on Smart Grid, 9 (2017), pp. 5535–5542.

[16] S. R. Shakeel, J. Takala, and W. Shakeel, *Renewable energy sources in power generation in pakistan*, Renewable and Sustainable Energy Reviews, 64 (2016), pp. 421–434.

[17] J. D. Smith, A. J. Hill, L. E. Reeder, B. C. Franke, R. B. Lehoucq, O. Parekh, W. Severa, and J. B. Aimone, *Neuromorphic scaling advantages for energy-efficient random walk computations*, Nature Electronics, 5 (2022), pp. 102–112.

[18] J. D. Smith, W. Severa, A. J. Hill, L. Reeder, B. Franke, R. B. Lehoucq, O. D. Parekh, and J. B. Aimone, *Solving a steady-state pde using spiking networks and neuromorphic hardware*, in International Conference on Neuromorphic Systems 2020, 2020, pp. 1–8.

[19] K. Tang, S. Dong, X. Ma, L. Lv, and Y. Song, *Chance-constrained optimal power flow of integrated transmission and distribution networks with limited information interaction*, IEEE Transactions on Smart Grid, 12 (2020), pp. 821–833.

[20] A. A. E. Tawfiq, M. O. A. El-Raouf, M. I. Mosaad, A. F. A. Gawad, and M. A. E. Farahat, *Optimal reliability study of grid-connected pv systems using evolutionary computing techniques*, IEEE Access, 9 (2021), pp. 42125–42139.

[21] B. Wang, P. Dehghanian, and D. Zhao, *Chance-constrained energy management system for power grids with high proliferation of renewables and electric vehicles*, IEEE Transactions on Smart Grid, 11 (2019), pp. 2324–2336.

[22] L. Yang, Y. Xu, W. Gu, and H. Sun, *Distributionally robust chance-constrained optimal power-gas flow under bidirectional interactions considering uncertain wind power*, IEEE Transactions on Smart Grid, 12 (2020), pp. 1722–1735.

[23] Z. Yang, R. Wu, J. Yang, K. Long, and P. You, *Economical operation of microgrid with various devices via distributed optimization*, IEEE Transactions on Smart Grid, 7 (2015), pp. 857–867.

[24] M. Zachar and P. Daoutidis, *Microgrid/macrogrid energy exchange: A novel market structure and stochastic scheduling*, IEEE Transactions on Smart Grid, 8 (2016), pp. 178–189.

[25] R. Zand, K. Y. Camsari, S. Datta, and R. F. DeMara, *Composable probabilistic inference networks using mram-based stochastic neurons*, ACM Journal on Emerging Technologies in Computing Systems (JETC), 15 (2019), pp. 1–22.

[26] R. Zand, K. Y. Camsari, S. D. Pyle, I. Ahmed, C. H. Kim, and R. F. DeMara, *Low-energy deep belief networks using intrinsic sigmoidal spintronic-based probabilistic neurons*, in Proceedings of the 2018 on Great Lakes Symposium on VLSI, 2018, pp. 15–20.

[27] H. Zhang, Z. Hu, E. Munsing, S. J. Moura, and Y. Song, *Data-driven chance-constrained regulation capacity offering for distributed energy resources*, IEEE Transactions on Smart Grid, 10 (2018), pp. 2713–2725.

[28] C. Zhao and R. Jiang, *Distributionally robust contingency-constrained unit commitment*, IEEE Transactions on Power Systems, 33 (2017), pp. 94–102.

Edited by: Rajanikanth Aluvalu
Special issue on: Evolutionary Computing for AI-Driven Security and Privacy: Advancing the state-of-the-art applications
Received: Dec 10, 2023
Accepted: Jan 4, 2024