Advanced Computing is at the Forefront of a New "Moonshot" Revolutionizing the North American Power Grid

Christopher S. Oehmen, *Pacific Northwest National Laboratory, Richland, WA, 99352, USA*

Slaven Peleš, *Oak Ridge National Laboratory, Oak Ridge, TN, 37830, USA*

Katarzyna Świrydowicz, *AMD, Austin, TX, 78735, USA*

Cosmin G. Petra, *Lawrence Livermore National Laboratory, Livermore, CA, 94550, USA*

Shrirang G. Abhyankar, *Pacific Northwest National Laboratory, Richland, WA, 99352, USA*

Wesley Jones and Matthew Reynolds, *National Renewable Energy Laboratory, Golden, CO, 80401, USA*

Robert C. Rutherford, *Pacific Northwest National Laboratory, Richland, WA, 99352, USA*

In the 50-plus years since humans first landed on the moon, computing has grown at breakneck speed. We are faced with another challenge that is just as daunting and just as important to overcome: modernizing the North American electric power grid. High-performance computing systems with specialized software will be an important element in rising to this challenge. We describe at a high level how software developed in the ExaSGD project addresses this "moonshot" goal by utilizing exascale computing and a novel high-performance solver software stack to support the mission of decarbonizing power grid operations in an environment of uncertain weather and climate. To reach the exascale benchmark, the team has made a number of first-of-their-kind innovations, including a novel method for stochastic optimization, fine-grained parallel methods for modeling power systems, and GPU resident sparse numerical linear solvers.

In the late 1960s, the world watched with great anticipation as NASA overcame challenge after challenge to safely transport humans to the moon and return them home, now known widely as the *moonshot*, and win the global space race. Although many of the challenges NASA faced were physical in nature, there was a new technology on the scene that brought many of those challenges within reach for the first time: computing. To support the moonshot goals, the world's fastest computers were conscripted to optimize designs for spacecraft components, assist in guidance and navigation, and help lead to insight about nearly every element of the missions.

Computing machines with the ability to perform math and logic faster than any human were an essential ingredient in the success of the moonshot, in part because time was of the essence. The nation that could solve the technical challenges first would be the one to win the "space race." Calculating the fastest with the help of computers was an essential element of solving problems quickly. But beyond the element of speed, there were other important elements for which computers represented an essential leap forward. The ability to calculate faster means that you get timely answers that are more *accurate*, that account for more *detail*, and which can be *repeated* many times to explore a wider design space in the same amount of time that it takes slower computers to get answers of lower quality. These other elements, accuracy, resolution, and

Digital Object Identifier 10.1109/MCSE.2024.3403029
Date of publication 29 May 2024; date of current version 30 September 2024.

uncertainty quantification, were just as essential in driving leaps in technology that led to huge advances in the human understanding required to achieve the moonshot. These same needs are present in many scientific and technical disciplines, from chemistry to astrophysics. So, world class computing continues to be a strategic investment and asset for the United States.

This story includes lessons learned about how new computational methods were required to take advantage of the power of computing at the exascale level to enable new insight into managing and planning for the future power grid. Specifically, the team had to overcome specific challenges in reformulating the traditional power grid models so that they could be expressed in a form that took advantage of improvements in computing hardware. This made it possible to expand grid models to account for weather and a much larger number of hypothetical failure modes (security constraints), both essential elements in modernizing the North American power grid.

A NEW MOONSHOT

In the 50-plus years since humans first landed on the moon, computing has grown at breakneck speed. We are faced with another challenge that is just as daunting and just as important to overcome: modernizing the North American electric power grid. High-performance computing (HPC) systems with specialized software will be an important element in rising to this challenge. There are multiple imperatives driving efforts to utilize HPC to address the challenges presented by modern power grids, including the following:

- *Climate imperative*: Decarbonizing the technologies that produce, transmit, and consume electricity will achieve an important decrease in net CO_2, which is a critical element of managing global climate change.
- *Economic justice and prosperity imperative*: Reinventing the power grid will make it possible to provide equitable access to resources and increase efficient delivery of power, which will improve people's quality of life and access to economic opportunities.
- *Reliable and resilient grid operations imperative*: Delivering power as reliably and safely as possible to points of use improves the reliability and resiliency of the entire U.S. critical infrastructure and leads directly to U.S. economic advantage.

Today's power grid is operated by a careful planning and management process that balances anticipated load (energy consumption) with that power that is generated. When the balance between load and generation is maintained through transmission and distribution networks, then power is reliably and safely delivered to consumers. This balance can be disturbed through weather events like hurricanes, disasters like fires or earthquakes, or through unanticipated changes in load (for instance, when an unexpected hot or cold snap occurs, prompting many people to turn on power-hungry heaters). In these cases, grid operators "contain" the disturbance using techniques such as *islanding*, which divides a network into disconnected, smaller subnetworks; or *load shedding*, which deliberately underserves some nonessential load through localized blackouts to prevent widespread collapse of the system.

As we address the challenge of modernizing the power grid, there are a lot of new elements in the system that introduce uncertainty and instability. New generation, storage, and transmission technologies must be integrated into the aging infrastructure of the current power grid. Efficient and reliable carbon-neutral or carbon-negative technologies generate power using solar, wind, sea, and other energy sources, which reduce dependence on fossil-fuel based generation. But these same technologies can also introduce a large amount of uncertainty into the grid because they depend on weather, which cannot be predicted perfectly. Furthermore, renewable power generation can experience large and abrupt ramping events, changing output generation in minutes or seconds. Thus, characterizing uncertainty, e.g., via high-fidelity uncertainty sets,[1] becomes critical for mitigating the effects of generation shortfalls and overproduction in systems with high penetrations of renewable energy resources.

Storing large amounts of power is one solution to "bridge the gap" when sudden changes in renewable generation create imbalances between generation and load. When generation drops suddenly, stored power can be used to make up the difference. When generated power suddenly exceeds what is being consumed, excess power can be saved by storing it until some future time when it is needed. Newly developed batteries and power electronics technologies increase the capacity for short-term storage of power, but they introduce their own uncertainties and instabilities into the grid.[2] Highly distributed storage includes concepts like electric cars, which can either provide power to or take power from an electric grid, depending on the need. But this has to be balanced with consumers' requirement to have a vehicle charged when it is needed and therefore introduces even more uncertainty.

Balancing generation and load in this highly dynamic and uncertain environment is a challenge that can be addressed with help from advanced computing applications. Similar to the moonshot programs, most of the power grid is built using physical components that do the main work of the system. Physical components on the power grid perform the task of moving power from generation sources to consuming endpoints. *Computing tasks* enable every aspect of how the power grid is monitored, operated, and controlled. And computers are used to find the most efficient ways to operate the grid given the balance of generation and load at a given time, and how it is predicted to change in the near future (usually in 30-min time slices). One example of this kind of calculation is optimal power flow (OPF) analysis, which determines the most efficient generation settings that will deliver power for an anticipated load. OPF calculations use a model that describes the interconnections between generation and load, as captured in a network diagram, and use the laws of physics to determine how power would flow through that system at minimal cost (or maximum efficiency) for that configuration.

Adapting OPF models for the modern power grid required us to account for several specific features of grid systems and implement them in a way that could harness the power of exascale computing. These include the following:

- *Security constraints*: For safe and reliable operations, OPF calculations are paired with contingencies known as *security constraints*. Performing OPF with security constraints (SCOPF) ensures that changes made to most efficiently operate the grid don't inadvertently make the grid fragile. Modeling security constraints is a way to contemplate loss of grid components and determine whether their loss leads to a cascading or otherwise catastrophic failure. SCOPF is often done by calculating what would happen in the grid if nothing changes (the base case), and pairing that with the collection of possible future states in which any single grid component fails. The collection of all grid models, each with a single element missing or broken, is called the "N−1 contingency" set. Per regulatory requirements, the power grid should be able to withstand failure of any single component. However, major events, such as hurricanes, often cause simultaneous failures of multiple grid components.[3] If operators would like to also ensure that the grid is safe and reliable when k components fail, then "N−k" calculations can be performed, but at great expense compared to the N−1 case. In fact, the number of calculations in SCOPF is *combinatorial* in k, where k is the number of simultaneous failures to be explored (k can be 1, 2, or a higher number). It follows from this fact that increasing awareness of multiple failures on the grid comes with a substantial growth in the number of computations that must be performed. As a result, increased awareness for safe operations in the power grid can easily require vast computing resources.

  The large collection of models with multiple concurrent failures can be simplified in some cases, such as weather-related damage. In this case, operators can focus the damage they are concerned about to a confined geographical space, thereby paring down the number of scenarios that must be solved. Similar shortcuts can be taken to focus SCOPF calculations on only the damage contingencies that are highly likely or highly impactful. But in all cases, increased computing capacity and speed makes it possible to see a larger space of possibilities, leading to better decision making, more safe and reliable operations, and cost savings.
- *Weather scenarios*: Extreme weather events are becoming more commonplace. The power grid increasingly relies on weather-based generation like wind turbines and solar farms, so weather scenarios are increasingly important in predicting how and where power can be generated. Understanding potential effects of weather on the grid increases the requirements for computing capacity. Consider a case where operators need to guarantee safe and reliable operations in an N−1 SCOPF calculation where wind turbines provide a significant amount of power. Because the amount of power provided by the turbines cannot be exactly known beforehand, the entire collection of calculations must be performed, each with a different wind generation profile. If chosen reasonably, this collection of wind generation profiles will give operators a good idea of what to expect in situations of high, low, and expected (or average) wind generation. If the ideal operating point of the grid also does not lead to catastrophic failure for the range of possible wind generation scenarios, then operators can be much more confident that they are making the right choices, even under uncertain wind conditions. But this means that all the security constraints must be considered for each weather scenario, significantly increasing the complexity of the SCOPF computation.
- *Power grid models*: The power grid can be represented using a graph wherein nodes are physical elements of the grid-like generation points,

load points, and buses; and edges represent transmission lines that connect the nodes. Power grids have two important features that dominate the mathematics of solving for their behavior. 1) Grid systems are extremely *sparse*, meaning that most elements are only connected to a small number of other elements, and 2) they are also *irregular*, meaning that structure in one part of a grid does not necessarily look like structure in another part of the grid. Numerically, these two features translate into significant challenges for using computers to model grid behavior because they prevent us from using many of the common computational science methods that normally make large calculations simple. This is described in more detail in the following section.

FINDING A SOLUTION

SCOPF calculations can be approached using several different techniques, but so far, only one technique (demonstrated by the ExaGO software package, leveraging solver technologies in HiOp)[4] has been shown to work at scales from a single laptop to the world's fastest supercomputer. As such, it gives stakeholders unprecedented flexibility for deployment in different settings. We focus here on the techniques used in that code. The computational workhorse of ExaGO is a linear solver that calculates the generation and line flows for the base case and contingencies, as described in the previous section. Linear solvers operate by arranging data in a large matrix so that a large system of linear equations can be solved. The connectivity structure of the grid is intuitively represented by a square matrix, where any nonzero entry describes a connection between physical elements. Because power grid systems are sparse, this results in a matrix that contains mostly "0"s. This creates numerical problems when performing matrix operations because most of the arithmetic is wasted. Because of its size, computing the matrix inverse is computationally infeasible. First, the inverse of a sparse matrix is often dense (which leads to a storage challenge for large values of N), and second, the number of operations needed to compute the inverse grows cubically with N (making it an extremely long process for large matrices, even on a supercomputer). Mathematical and computational approaches that respect matrix sparsity properties and use them to their advantage are needed. Mathematical algorithms, known as *sparse direct* and *iterative linear solvers*, are examples of such approaches.

Because power grids are irregular leads to a second challenge with the matrix representation. Irregularity leads to a situation where nonzero entries do not follow any simple structure, like block-diagonal or banded patterns. For other applications where matrices have structure in their nonzero entries, those structures can often be exploited to streamline operations on those matrices. The regular matrices can often be stored efficiently (i.e., as a collection of diagonals). But in the case of the power grid, sparseness and irregularity mean that the native representations of grids are best served by a general sparse representation. In a sparse format, only nonzero entries are used, but each entry needs to be paired with information about where it should reside in the matrix. For very sparse systems such as the power grid, sparse representations require much less memory, so they can accommodate much larger systems than can dense representations. Further, many fewer mathematical operations are performed (for instance, when multiplying the matrix by a vector), so wall clock times are typically better. But the tradeoff is that many more logic and memory moving operations are performed because the matrix operations must constantly be checking where in memory other elements are located that need to be combined with a given element. So, benchmarks of hardware utilization that count only floating-point arithmetic frequently measure lower performance efficiency, even though the time to solution can be vastly improved.

The next layer moving outward is a nonlinear solver that uses the output of the linear solver to direct its efforts across a complex solution space, using the value of a fitness function (in our case, cost) to guide solutions toward ever-improved fitness. The logic and control in the nonlinear optimizer determine the number of iterations required to find a solution, the convergence criteria (i.e., what has to be true for the software to believe that it has found a solution), and how it interprets the results of calls to the linear solver to continue to march toward an optimal solution. ExaGO supports two different nonlinear optimizers: HiOp,[4] developed within the ExaSGD project, and legacy package Ipopt.[5] Nonlinear optimization can be used to find optimal solutions for many different application domains.

The third layer outward is software that resolves details that are specific to the power grid. It is this layer that ensures that the solutions being sought by the nonlinear optimizer reflect the actual physics and constraints of the power grid being analyzed. This includes details of how generation happens and how it can change, constraints of transmission lines, and balancing load and generation across the system.

The outermost layer of the software stack, called *HiOp PriDec*, spreads pieces of the calculation across systems that have multiple compute nodes equipped with different processing units. This layer uses primal decomposition to break the large calculation down into independent pieces that can be sent to different computing elements, and then reassembles the answers into a global solution. The details of this method developed within ExaSGD are presented in Wang and Petra.[6]

POWER OF COPROCESSOR ACCELERATORS

Keeping pace with the staggering improvements in computing power for the last six decades has required innovation at every step. Until recently, improvements in computational technology followed Moore's law, which states that the number of transistors in a microchip will double every two years. Moore's law is not really a law of science but rather a projection of how semiconductor technology development will grow, and that projection has been accurate for nearly 50 years. Miniaturizing transistors at such an exponential rate helped to dramatically increase computer memory and processing capability per unit of power supply. At this point in time, we have reached the physical limitations of semiconductor materials. Transistors at the subnanometer scale would be affected by quantum uncertainties and would not be able to act as reliable components in a processing device. Furthermore, at clocking frequencies of above 10 GHz, quantum effects would also affect reliability of computations and, as a result, we cannot expect computer processors to become much faster than they are today. To improve processing power beyond Moore's law, we need to find a way to do more computations with fewer transistors. The most recent wave of innovation in this direction has seen an explosion in the amount of computing power that resides in massively parallel accelerators such as GPUs. This trend is revealed in the top 500 computer list, the top of which is dominated by processing power that resides in accelerators and not in traditional CPUs. Historically, the world's fastest supercomputers have been bleeding-edge indicators of trends that find their way into commodity computing platforms, so software that is refactored to run on high-end systems is ahead of the curve for running on future off-the-shelf systems.

Although accelerator devices deliver high processing power per unit of power supply, they are not as flexible as CPUs. Mathematical algorithms need to satisfy additional constraints to harvest GPU computational power. Applications that natively see the greatest benefit from accelerators are applications that 1) perform the same instruction sequence on as many different data as possible on as many of those data as possible, 2) divide data into independent regions that can be operated on simultaneously and store their results in independent memory locations, 3) rarely move data back and forth between the CPU and the accelerator; and 4) fill the accelerator with dense data. SCOPF does *not* natively fit this pattern because our problem is most natively sparse and irregular. Furthermore, after an exhaustive survey,[7] we found that there were no off-the-shelf *sparse* linear solvers that we could use effectively on GPUs. To address this, we took significant steps to "densify" the power grid representation as much as possible, which improves utilization of the GPU coprocessors at the cost of less favorable scaling properties, as listed in Tables 1 and 2. This mixed dense/sparse representation, which we detail in Abhyankar et al.,[8] was used to create the first-ever exascale SCOPF application, which was demonstrated to run on the Frontier supercomputer at Oak Ridge National Laboratory (ORNL).

Three features drive the scale-of-computing SCOPF calculations used by power grid operators and planners: 1) the number of elements (i.e., buses, lines, and generators) in the grid model being used, 2) the number of contingencies to be considered for the grid model, and 3) the number of weather scenarios to be used. Table 1 illustrates the scale of computing needed for different combinations of these three elements for mixed dense/sparse systems.

POWER GRID MODERNIZATION MOONSHOT

In late April and early May of 2023, the ExaSGD team performed the first exascale calculations for power grid optimization (SCOPF) using the ORNL system Frontier, the world's first exascale system. These calculations were performed on synthetic but representative grid models that capture the essential features of a system the size of the Western U.S. grid. When compared to industry-standard calculations for the same system, ExaGO and HiOp running on Frontier was able to explore a much larger space of possible operational modes as well as 10 weather scenarios. Being able to "see" this larger landscape made it possible to find an optimal solution with respect to an ensemble of different contingencies and weather scenarios, which describe uncertainties in decarbonized grid operations. No industry tool today is capable of performing analysis at such scale.

TABLE 1. Scale of computing required to perform *mixed dense/sparse* SCOPF calculations with multiple weather scenarios.

Grid model size	Number of contingencies	Number of weather scenarios	Scale of computing required
Local, 200 buses (∼500	500 (N−1)	1	Gigascale: laptop
total elements)	500 (N−1)	100	Terascale: single CPU with GPU
	250,000 (N−1)	10	accelerators
Regional grid, 2000 buses	6000 (N−1)	100	Petascale: HPC cluster or a cloud
(∼6000 total elements)	36×10^6 (N−1)	10	resource.
Western interconnection,	23,000 (N−1)	10	Exascale: leadership class
10,000 buses (∼23,000 total			computing system
elements)	5.3×10^8 (N−1)	10	Beyond exascale
Eastern interconnection,	170,000 (N−1)	100	
70,000 buses (∼170,000	29×10^9 (N−1)	10	
total elements)			

SPARSE COMPUTING

The ExaSGD team has followed this success by developing a next-generation SCOPF solver that utilizes a sparse representation that more natively matches the underlying grid model, as opposed to using the mixed dense/sparse approach described earlier. This has increased the size of grid model that can be analyzed, and it has reduced the time it takes for the computations to unfold. To do this, the team, along with collaborators, developed new math libraries that perform sparse matrix operations—in particular, linear solvers—on accelerators.[9,10] These software libraries have been used to refactor the ExaGO and HiOp codes so that they can take advantage of accelerators without having to force fit the underlying mathematics and grid models into a mixed dense/sparse representation. This has several implications that immediately improve the performance of SCOPF calculations.

First, working within a sparse representation allows ExaGO to operate on much larger grid systems on a single accelerator than were possible before. The impact of this change is that grid models that capture the entire North American power grid now can fit on a single accelerator.

Second, as expected, streamlined sparse math libraries have demonstrated even faster runtimes than were observed using the mixed dense/sparse approach. This means that ExaGO would enable grid operators and planners to compute decisions using even more weather scenarios and possible failure modes than possible with either state-of-the-art tools, or even ExaGO using the mixed dense/sparse approach. More scenarios and failure modes lead to better awareness for grid operators. Better awareness translates to safer and more reliable operation of the grid at a lower cost.

Taken together, larger grid models and faster runtimes mean that supercomputer performance is within reach of grid operators on commodity systems. Table 2 illustrates how the challenging problems in Table 1 are more accessible when using the sparse linear solvers we developed.

IMPACT

As illustrated in the 2021 Midcontinent Independent System Operator Renewable Integration Impact Assessment,[11] once the penetration of renewable generation exceeds 30%, the complexity of the ensuing power grid

TABLE 2. *Sparse* solver brings larger calculations within reach.

Grid model size	Number of contingencies (x) weather scenarios	Mixed dense/sparse performance (from Table 1)	*Improved* sparse performance
Local, 200 buses (∼500 total	500	Gigascale: laptop	Gigascale: laptop
elements)	500,000–2,500,000	Terascale: single node with	
		GPU accelerators	
Regional grid, 2000 buses	600,000	Petascale: HPC cluster or	Terascale: single node
(∼6,000 total elements)		cloud	with GPU accelerators
	360,000,000	Exascale: leadership class	Petascale: HPC cluster
Western interconnection,	230,000	computing system	or a cloud resource
10,000 buses (∼23,000 total	53×10^8	Beyond exascale	Exascale: leadership
elements)			class computing system
Eastern interconnection,	17,000,000		
70,000 buses (∼170,000 total	2.9×10^{11}		Beyond exascale
elements)			

grows exponentially, which will require "significant changes to current operating, market, and planning practices." Achieving the moonshot goal of a modernized power grid will be enabled by advances in computing at all scales. The advances described herein will advance understanding of the small grids, microgrids, or segments managed by single operators using commodity hardware, allowing users at this scale to have awareness of how their grids will behave in a massively more complex engineering environment. These same algorithmic advances enable whole-grid-scale awareness in the context of multiple failures and complex interactions with weather and climate, making it possible to further optimize safe and reliable delivery of power to the nation's consumers even as the complexity of the grid continues to grow.

USING WHAT WE LEARNED TO ADVANCE OTHER SCIENCES

The lessons learned along the journey as the first SCOPF exascale application are key steps in helping modernize the North American power grid in the presence of uncertainty brought on by the complexities of modern power generation through renewables, battery storage (see, e.g., Satkauskas et al.),[12] and the electrification of U.S. automobiles. Algorithm improvements have brought enormous calculations within reach of grid operators who do not have access to exascale systems, but who could benefit from modest multiprocessor systems. But the benefits of this research do not stop there. The power grid is mathematically similar to other complex systems. For instance, the way that nutrients and chemicals or even control signals flow through complex biological systems can also be represented using sparse irregular models. Other engineered systems that include complex control and organization (such as those critical for the success of Energy Earthshots) also have a similar representation. And for all of these systems, there are applications to optimize them. For biosystems, you might want to add features to produce clean energy or provide resistance to bio threats. When designing engineered systems, one wants to optimize their efficiency while minimizing the product development and the operational cost. The key is that these other application areas are mathematically similar to how we represent the power grid. These other systems are irregular and sparse, and the goal is to optimize them in terms of robustness or similar metrics. The problems that can be mapped into such a formulation will benefit from the computational advances herein applied to power grid optimization. So, the lessons learned and software developed for performing exascale calculations on Frontier through the Exascale Computing Program have resulted in a larger contribution to science broadly to take advantage of advanced computing platforms of all sizes.

ACKNOWLEDGMENTS

This research was supported by the Exascale Computing Project (17-SC-20-SC), a joint project of the U.S. Department of Energy's (DOE's) Office of Science and National Nuclear Security Administration responsible for delivering a capable exascale ecosystem, including software, applications, and hardware technology, to support the nation's exascale computing imperative. 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. This work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the DOE under Contract DE-AC36-08GO28308.

REFERENCES

1. K. Panda, I. Satkauskas, J. Maack, D. Sigler, M. Reynolds, and W. Jones, "A data-driven multi-period importance sampling strategy for stochastic economic dispatch," in *Proc. IEEE Power Energy Soc. General Meeting (PESGM)*, Piscataway, NJ, USA: IEEE Press, 2021, pp. 1–5, doi: 10.1109/PESGM46819.2021.9638215.
2. M. Paolone et al., "Fundamentals of power systems modelling in the presence of converter-interfaced generation," *Elect. Power Syst. Res.*, vol. 189, Dec. 2020, Art. no. 106811, doi: 10.1016/j.epsr.2020.106811.
3. I. Satkauskas, J. Maack, M. Reynolds, D. Sigler, K. Panda, and W. Jones, "Simulating impacts of extreme events on grids with high penetrations of wind power resources," in *Proc. IEEE/PES Transmiss. Distrib. Conf. Expo. (T&D)*, Piscataway, NJ, USA: IEEE Press, 2022, pp. 01–05, doi: 10.1109/TD43745.2022.9816884.
4. C. G. Petra, "A memory-distributed quasi-Newton solver for nonlinear programming problems with a small number of general constraints," *J. Parallel Distrib. Comput.*, vol. 133, pp. 337–348, Nov. 2019, doi: 10.1016/j.jpdc.2018.10.009.
5. A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2006, doi: 10.1007/s10107-004-0559-y.
6. J. Wang and C. G. Petra, "A sequential quadratic programming algorithm for nonsmooth problems with

upper-C^2 objective," *SIAM J. Optim.*, vol. 33, no. 3, pp. 2379–2405, 2023, doi: 10.1137/22M1490995.
7. K. Świrydowicz et al., "Linear solvers for power grid optimization problems: A review of GPU-accelerated linear solvers," *Parallel Comput.*, vol. 111, Jul. 2022, Art. no. 102870, doi: 10.1016/j.parco.2021.102870.
8. S. Abhyankar, S. Peles, R. Rutherford, and A. Mancinelli, "Evaluation of AC optimal power flow on graphical processing units," in *Proc. IEEE Power Energy Soc. General Meeting (PESGM)*, Washington, DC, USA, 2021, pp. 1–5, doi: 10.1109/PESGM46819.2021.9638131.
9. K. Świrydowicz et al., "GPU-resident sparse direct linear solvers for alternating current optimal power flow analysis," *Int. J. Elect. Power Energy Syst.*, vol. 155, Jan. 2024, Art. no. 109517, doi: 10.1016/j.ijepes.2023.109517.
10. S. Regev et al., "HyKKT: A hybrid direct-iterative method for solving KKT linear systems," *Optim. Methods Softw.*, vol. 38, no. 2, pp. 332–355, 2023, doi: 10.1080/10556788.2022.2124990.
11. Midcontinent Independent System Operator, "MISO's renewable integration impact assessment," MISO, Carmel, IN, USA, Rep. 520051, 2021.
12. I. Satkauskas, J. Maack, M. Reynolds, D. Sigler, K. Panda, and W. Jones, "Simulating impacts of extreme events on grids with high penetrations of wind power resources," in *Proc. IEEE/PES Transmiss. and Distribution Conf. Expo. (T&D)*, New Orleans, LA, USA: IEEE Press, 2022, pp. 1–5, doi: 10.1109/TD43745.2022.9816884.

CHRISTOPHER S. OEHMEN is a research scientist at Pacific Northwest National Laboratory, Richland, WA, 99352, USA, where he leads the Computational Biology group in the Biological Sciences Division. His research interests include computational biology, bioinspired cybersecurity, and high-performance computing for complex systems. Oehmen received his Ph.D. degree in biomedical engineering from the University of Memphis/University of Tennessee, Health Science Center Joint program. Contact him at chris.oehmen@pnnl.gov.

SLAVEN PELEŠ is a senior scientist with the Computational Systems Engineering and Cybernetics group, Oak Ridge National Laboratory, Oak Ridge, TN, 37830, USA. His research interests include model-based and data-driven computational methods for the design of power systems, microgrids, and building energy systems. Peles received his Ph.D. degree in physics from the University of Manitoba. Contact him at peless@ornl.gov.

KATARZYNA ŚWIRYDOWICZ is a senior member of the technical staff at AMD, Austin, TX, 78735, USA. Her research interests include linear solvers and computational linear algebra, and performance optimization for GPU-based implementations. Swirydowicz received her Ph.D. degree in mathematics from Virginia Tech. Contact her at kasia.swirydowicz@pnnl.gov.

COSMIN G. PETRA is a computational mathematician at Lawrence Livermore National Laboratory, Livermore, CA, 94550, USA. His research interests include high-performance computing algorithms and solvers for mathematical optimization problems, with an emphasis on the operation, design, and planning of complex energy systems. Petra received his Ph.D. degree in applied mathematics from the University of Maryland, Baltimore County. Contact him at petra1@llnl.gov.

SHRIRANG G. ABHYANKAR is a senior scientist in the Optimization and Control group, Pacific Northwest National Laboratory, Richland, WA, 99352, USA. His research interests lie at the intersection of power system simulation, numerical methods, and large-scale computing. Abhyankar received his Ph.D. degree in electrical engineering from the Illinois Institute of Technology. He is a Senior Member of IEEE. Contact him at shrirang.abhyankar@pnnl.gov.

WESLEY JONES is a principal scientist and group manager at the National Renewable Energy Laboratory, Golden, CO, 80401, USA. His research interests include sustainable energy, high-performance computing, and quantum computing. Jones received his Ph.D. degree in physics from Dartmouth College. He is a Member of IEEE. Contact him at wesley.jones@nrel.gov.

MATTHEW REYNOLDS is an applied mathematics researcher at the National Renewable Energy Laboratory, Golden, CO, 80401, USA. His research interests include machine learning, uncertainty quantification, and optimization under uncertainty. Reynolds received his Ph.D. degree in applied mathematics from the University of Colorado Boulder. Contact him at matthew.reynolds@nrel.gov.

ROBERT C. RUTHERFORD is a research software engineer in the High Performance Computing, Machine Learning and Quantum Computing group, Research Computing, Pacific Northwest National Laboratory, Richland, WA, 99352, USA. His research interests include development of low-level linear algebra kernels and software engineering support for a variety of platforms. Rutherford received his bachelor of science degree in mathematics and computer science from Whitworth University. Contact him at robert.rutherford@pnnl.gov.