[[1]](#footnote-1)

Usability Improvements for Physical Simulations of Electric Grids Using MATPOWER

First A. Author, *Fellow, IEEE*, Second B. Author, and Third C. Author, Jr., Member, IEEE

*Abstract*—Physical simulations of electric grids using MATPOWER provide critical information about cascading failures based on the physical attributes of the grids [1]. While previously developed tools allow for physical simulations on smaller grids in reasonable timeframes for research purposes, they fail to scale to larger grids. Furthermore, the single-threaded simulation tools cannot complete simulations for real-time use cases, such as for electrical grid operators [2]. Parallelization improvements made to the physical simulator allow for real-time, accurate, and larger scale simulations of cascading failures using multi-core workstations and supercomputers. A new GUI provides an accessible front-end to the physical simulators developed by Marquette’s SACE lab for use within the lab, by other interested labs, and electric grid operators. The goal of this paper is to quantify the performance improvements to physical simulations, explain how they were achieved, and provide instruction on the usage of the new GUI and CLI.

*Index Terms*—Enter key words or phrases in alphabetical order, separated by commas. For a list of suggested keywords, send a blank e-mail to [keywords@ieee.org](mailto:keywords@ieee.org) or visit <http://www.ieee.org/organizations/pubs/ani_prod/keywrd98.txt>

# INTRODUCTION

Cascading failures are electrical grid events where line-failures, instigated by an attacker or naturally occurring, spread to more lines, with the possibility of leading to full blackouts without proper intervention. Cascading failures which lead to blackouts often have enormous monetary and societal costs, such as the North American blackouts [3]. With limited options for avoiding cascading failures, the aim is instead to limit the spread of cascading-failure events

Physical simulations of electric grids using MATPOWER provide critical information about cascading failures based on the physical attributes of the grids.

(What is a cascading failure? Why is there a simulator?)

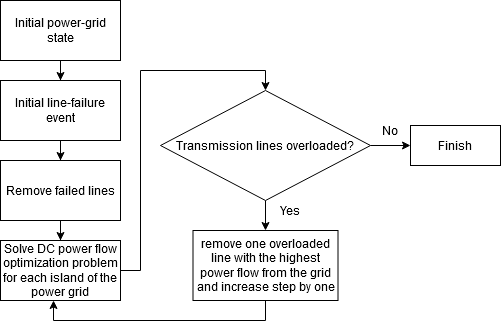
While previously developed tools allow for physical simulations on smaller grids in reasonable timeframes for research purposes, they fail to scale to larger grids, with IEEE300 simulations taking more than 10 minutes on an i7 at 4GHz [2]. This simulation time shows that single-threaded simulation tools cannot complete simulations for real-time use cases, such as for electrical grid operators, where failure events often occur in the span of a few minutes [1]. Parallelization improvements made to the physical simulator allow for real-time, accurate, and larger scale simulations of cascading failures using multi-core workstations and supercomputers. A new GUI provides an accessible front-end to the physical simulators developed by Marquette’s SACE lab for use within the lab, by other interested labs, and electric grid operators. The goal of this paper is to quantify the performance improvements to physical simulations, explain how they were achieved, and provide instruction on the usage of the new GUI.

Currently, using the simulator requires understanding of the MATLAB code and manipulation of variables within the code, making it inaccessible to outside labs without documentation and grid operators for on-field use. The current simulator code relies on dependencies not packaged with the code, and thus is not portable. The GUI package fixes this by including all dependencies, allowing for use on any system capable of running MATLAB R2020a.

On top of this, the tools fail to complete large simulations in reasonable times for real-time applications while also failing to complete simulations of larger grids in reasonable times for research within the SACE lab. This paper and the corresponding simulation tools and simulation improvements opens the possibility of simulating larger grids and future works, such as a web tool, while also presenting grid information in an understandable format for research work and real-time use. A simple interface allows users to change the operating characteristics of the grid without prior coding knowledge and provides both information in a form for experienced users (a state matrix) and for new users (graphs).

The simulator tool currently works with both the IEEE 118 and IEEE 300 bus electric grid topologies, both of which provide the greatest stability and lowest complexity with outputs that scale up to larger grids. Currently available simulation tools fail to simulate larger grids, but the parallelization added with this tool means that with modifications, reliable and fast simulation could be performed on more complex topologies using workstations and supercomputers.

**Fig. 1** shows a simple model of the cascading failure simulation using MATPOWER and modifications for removing lines and creating islands used in [1]. Each cascading failure starts with the failure of a specified number of lines, which then requires DC optimization of the grid and any islands formed. After DC optimization, the simulator checks if any lines are overloaded, and then removes the overloaded line with the highest power flow, increasing the step by one, or if no lines are overloaded, finishes the simulation. If a line is removed, DC optimization and the transmission line failure check are performed again, with the loop continuing until no transmission lines overload as a result of DC power flow optimization.



(Difficulties a user faces without a GUI/CLI)

(Not everyone is competent at coding)

(Currently need to change the MatLab code)

(Web tool? – future work)

(Entire page)

(What are the requirements?)

(Advantages?)

(Assume user doesn’t know anything about cascading failure and work)

(Include definition of cascading failure)

(Impacts of cascading failures)

(Examples of cascading failures)

(IEEE grids I used)

(Rewrite things from other papers)

(What are the parameters?)

(Why 118 and 300 models. How do they scale to full US grid?)

(1 page summary of works done previously at our research lab)

(Give details about MATPOWER simulation)

(1 page – what additions have been made to existing MATPOWER simulator (From previous papers) – how it was turned into cascading failure simulator)

(Time complexity – how the time scales)

(This simulator vs. old simulator)

(Include human error probability column)

(Check Rezoan’s paper – Modeling Cascading Failures using Machine Learning algorithms – 2019 NAPS paper)

(Human Error paper)

(Go through abstracts of all papers)

# Related works

[1] provides a scalable model for analyzing cascading failures in power grids using a continuous-time Markov chain using state-variables and the operating characteristics of the grid, such as load-shedding allowance, load demand, and capacity estimation error. Using the states’ variables, functions such as probability mass of blackout size and cascading failure-stop probability are calculated.

[2] describes a process of using a Markov decision process (MDP) previously used in [1] to develop load-shedding policies to mitigate cascading failures and validates the results using physical electric grid cascading failure simulations based on optimal DC power flow.

The simulator used in this paper accounts for human operator actions and error and was originally developed for [4].

The purpose of development for this GUI tool is very similar to the tool developed in [5]. Like [5], the purpose of this tool is to transform raw data, a state matrix, into useful information through graphs, and allow for saving it into an Excel spreadsheet.

(Analytical Model)

(2014 paper)

(Human operator paper)

(Combination of human error + communication)

(Describe some other GUI-based paper – 1 paragraph)

(Go through abstract and names of paper – not the whole thing)

# Parallelization

[2] provides performance figures for physical simulations of cascading failures of power-grids using the backwards induction algorithm in MATLAB on a single thread of a Core i7 CPU at 4Ghz. While this provides reasonable times for smaller grids and few cascading failures (lower accuracy), when scaled up to larger grids and larger numbers of cascading failures, the algorithm fails to provide reasonable times for computing real-time actions. A parallel processing improvement is mentioned, but not explored.

Each cascading failure runs independent of other cascading failures, only depending on the initial state of the power grid and initial line failure events. To best form conclusions using the physical simulator, many iterations (1000+) are required. The self-contained nature of the cascading failure simulations means that with parallel-for loops and a data structure designed for parallel workloads, such as a cell matrix, multiple cascading failures can run in parallel. Simulation work consists almost entirely of cascading failures, meaning that running *x* cascading failures in parallel leads to simulations speeds *x* times faster than single-threaded performance (when a processor runs at the same speed).

The new physical simulator enables cascading failures to run completely in parallel, allowing for scalability to multi-core workstations and supercomputers, and opening the possibility of GPU-processing to accelerate simulations further. Fig. 1 shows the original times for 5000 iterations and 100 iterations respectively on the IEEE 118 and IEEE 300 bus systems with operating conditions included using a version of the physical simulator that includes human-error-probability. As shown, even the use of parallelization on a consumer 8-core AMD R7 1700 at 4Ghz running 8 cascading failures at a time opens opportunities for real-time simulations on larger power grids, such as the IEEE 300 bus grid, while allowing for more iterations on smaller grids, such as the IEEE 118 bus grid.

As shown by Fig. 1, simulation speed scales linearly with number of cores, each thread allowing for another cascading failure to be run in parallel. This means that many-core computers such as workstations, which range up to 64 cores, servers, and supercomputers could handle far more iterations of larger grids by running more cascading failures in parallel.

The single-threaded simulator cannot run simulations of larger grid topologies, such as the topology of the polish grid, within reasonable times for testing and confirmation purposes, limiting the confidence in the applicability of conclusions made using simulations of smaller grids. By providing a simulator that scale to large supercomputers, such as those on college campuses, the parallelized simulator allows for greater confidence in results and faster feedback on the functionality of new features in the simulator with larger grids, which can often cause unforeseen errors.

# GUI

The GUI provides an easy-to-use interface for visualizing cascading failure simulations via cascading failure probability, blackout-size distribution, and any other graphs that use state data from the electric grid. The GUI’s purpose is to provide a front-end for use in creating, presenting, and saving cascading failure state data for electric grids. The GUI should be usable and modifiable for both research purposes and use in the field.

Fig. 2 shows the layout of the GUI. The GUI allows the user to directly change the operating characteristics of the grid using the numerical inputs or to load in a previously generated state matrix. Included are graphs to visualize the state matrix, the state matrix itself, and the ability to save the state matrix to an Excel file.

# How to use

The parallelization update requires MATLAB’s deep learning and parallelization toolboxes, making use of the cell matrix and parallel-for loop functionality. For DC optimization of the electric grid, MATPOWER 7.0 is used, thus it must be installed in the MATLAB session. A packaged MATLAB app is available at <https://github.com/kassieclaire/Physical-Simulator-GUI-Package> .

Fig. 3 shows a detailed diagram of the GUI. When the original operating characteristics of the grid are known, the user can input them and start a simulation by entering the number of iterations (cascading failures) and clicking the simulate button. If the user already has a state matrix which they want to analyze, they can input the grid topology from the drop-down menu and the name of the MATLAB-compatible spreadsheet in the fill-in box next to the “Analyze Matrix” button.

References (at least 15 – 1 column)

1. M. Rahnamay-Naeini, Z. Wang, N.Ghani, A. Mammoli , and M. M. Hayat, “[Stochastic Analysis of Cascading-Failure Dynamics in Power Grids, IEEE Transactions on Power Systems](https://www.google.com/url?q=https%3A%2F%2Fieeexplore.ieee.org%2Fdocument%2F6714578&sa=D&sntz=1&usg=AFQjCNGTez_SDunQuUihtJFMuMeuH18E_w),” In IEEE Transactions on Power Systems, vol.29, no.4, pp.1767-1779, July 2014.
2. (Insert the paper that includes performance figures here)
3. P. Hines, J. Apt, and S. Talukdar, “Large blackouts in north america: Historical trends and policy implications,” Energy Policy, vol. 37, pp. 5249–5259, 2009.
4. (human operator paper)
5. (NMDOT GUI Paper (Hayat)))

**First A. Author** (M’76–SM’81–F’87) and all authors may include biographies. Biographies are often not included in conference-related papers. This author became a Member (M) of IEEE in 1976, a Senior Member (SM) in 1981, and a Fellow (F) in 1987. The first paragraph may contain a place and/or date

1. This paragraph of the first footnote will contain the date on which you submitted your paper for review. It will also contain support information, including sponsor and financial support acknowledgment. For example, “This work was supported in part by the U.S. Depart­ment of Com­merce under Grant BS123456.”

   The next few paragraphs should contain the authors’ current affiliations, including current address and e-mail. For example, F. A. Author is with the National Institute of Standards and Technology, Boulder, CO 80305 USA (e-mail: author@ boulder.nist.gov).

   S. B. Author, Jr., was with Rice University, Houston, TX 77005 USA. He is now with the Department of Physics, Colorado State University, Fort Collins, CO 80523 USA (e-mail: author@lamar.colostate.edu).

   T. C. Author is with the Electrical Engineering Department, University of Colorado, Boulder, CO 80309 USA, on leave from the National Research Institute for Metals, Tsukuba, Japan (e-mail: author@nrim.go.jp). [↑](#footnote-ref-1)