



Predicting Smart Grid Stability with Optimized Deep Models

Paulo Breviglieri¹ · Türkücan Erdem² · Süleyman Eken²

Received: 28 November 2020 / Accepted: 9 January 2021 / Published online: 4 February 2021
© The Author(s), under exclusive licence to Springer Nature Singapore Pte Ltd. part of Springer Nature 2021

Abstract

In a smart grid, consumer demand information is collected, centrally evaluated against current supply conditions and the resulting proposed price information is sent back to customers for them to decide about usage. As the whole process is time dependent, dynamically estimating grid stability becomes not only a concern but a major requirement. Decentral Smart Grid Control (DSGC) systems monitor one particular property of the grid—its frequency. So, it ties the electricity price to the grid frequency so that it is available to all participants, i.e., all energy consumers and producers. DSGC has some assumptions to infer the behavior of participants. DSGC system is described with differential equations. In this paper, we study on optimized deep learning (DL) models to solve fixed inputs (variables of the equations) and equality issues in DSGC system. Therefore, measuring the grid frequency at the premise of each customer would suffice to provide the network administrator with all required information about the current network power balance, so that it can price its energy offering—and inform consumers—accordingly. To predict smart grid stability, we use different optimized DL models to analyze the DSGC system for many diverse input values, removing those restrictive assumptions on input values. In our tests, DL model accuracy has reached up to 99.62%. We demonstrate that DL models indeed give way to new insights into the simulated system. We have learned that fast adaptation generally improves system stability.

Keywords Smart grid stability · Deep learning · Optimized learning rate · Grid stability prediction · Hyperparameter optimization

Introduction

The rise of renewable energy provides a cleaner, much needed and wanted an alternative to fossil fuels. Adaptation of renewable energy is not without its issues, especially two interrelated issues that are worthy of extra attention. Before the advent of renewable energy, traditional grids consisted

of few production centers that supplied energy to consumers and featured unidirectional flows between. However, with the ascent of renewable came a mix of these two called “prosumers”, these prosumers both consume and supply energy. This requires energy flow in grids to become bidirectional. The emergence of these “prosumers” and increased that comes with renewable energy also comes with the more complex generation, distribution and consumption. Economic implications tied to this, especially the choice of whether to buy or not to buy energy at given price, have too become more complex and challenging [1]. Relevant contributions on how to tackle the requirements of such a new scenario have been offered by the academy and industry over the past years. Special attention has been devoted to the study of smart grid stability [2].

Smart grids work by collecting information about consumer demand, evaluating this information against current supply information, finding a price value for the electricity and sending this price information to customers so that they can decide about their usage. Since, the process is time dependent, dynamically estimating grid stability is one of

This article is part of the topical collection “Cyber Security and Privacy in Communication Networks” guest edited by Rajiv Misra, R K Shyamsunder, Alexiei Dingli, Natalie Denk, Omer Rana, Alexander Pfeiffer, Ashok Patel and Nishtha Kesswani.

✉ Süleyman Eken
suleyman.eken@kocaeli.edu.tr

Paulo Breviglieri
info@paulobreviglieri.com

Türkücan Erdem
195172019@kocaeli.edu.tr

¹ Data Scientist, sciQant LLC, São Paulo, Brazil

² Department of Information Systems Engineering, Kocaeli University, 41001 İzmit, Turkey

the most important requirements of such a system. To analyze disturbances and fluctuations in energy consumption/production in a dynamic way not only the technical aspect of the system but the interrelated economic aspect of energy prices also must be taken in to account.

DSGC tries to tackle this by taking advantage of a property of the power grid. In times of excess generation frequency of the power grid increases and likewise in times of underproduction frequency of the power grid decreases. This means all the information needed for finding energy prices can be gotten from measuring frequency [3]. Therefore, measuring the grid frequency at the premise of each customer would suffice to provide the network administrator with all required information about the current network power balance, so that it can price its energy offering—and inform consumers - accordingly.

The DSGC differential equation-based mathematical model described in [3] and assessed in [5] aims at identifying grid instability for a reference four-node star architecture, comprising one power source (a centralized generation node) supplying energy to three consumption nodes. The model takes into consideration inputs (features) related to the total power balance (nominal power produced or consumed at each grid node); the response time of participants to adjust consumption and/or production in response to price changes (referred to as reaction time); energy price elasticity.

So we have a mathematical model with which grid instability can be predicted. The need for a tool to predict grid instability would have been met, and the binary classification (“stable” versus “unstable”) problem would be solved. However, the execution of this model relies on significant simplifications. A differential equation-based model can be manipulated in several ways. One traditional approach consists of running simulations with a combination of fixed values for one subset of variables and fixed value distributions for the remaining subset. As elegantly depicted in [4], this strategy leads to two primary issues, referred to as the “fixed inputs issue” and the “equality issue”. Please refer to [4] for a comprehensive assessment of both issues. Contributions to the literature with the paper can be listed as follows:

- Optimized deep learning models proved to be an outstanding prediction tool for smart grid stability. Even considering that the dataset is well behaved and needed no significant preprocessing, the high accuracy values obtained on the testing set confirm that a deep learning model may be safely considered.
- The use of an augmented dataset with 6.000 observations contributed significantly to better results.

The remainder of this article is organized as follows. In the next section, relevant works for smart grid stability are given. The subsequent section presents material and methods

followed by which, experimental results on the original and augmented simulation dataset are given. The last section concludes the article and gives future works.

Related Works

Alternative approaches have been proposed to overcome the inherent DSGC model simplifications. Venayagamoorthy [3] introduced situational awareness (SA) in the context of stability. SA is a perception of elements in the environment in a given time and space. It is critical for secure and efficient smart grid operation. Gaining more info does not reduce uncertainty, concerns reside with info sensed. For desired intelligent systems there are; sense-making agents, decision-making agents and adaptation agents with a knowledge base at center. Intelligent sense making is required for complex situations with time constraints. For intelligent sense-making; neural networks, fuzzy logic swarm intelligence and adaptive critic designs are found to be promising. The fuzzy logic-based approach is also found to be promising for judging network stability. Wide Area Monitoring (WAM) is found to be essential. With more interconnections and plug and play elements behaving as sinks or sources advanced monitoring systems learning functions, including cellular neural networks and stochastic identifiers are found to be needed.

Arzamasov et al. [4] proposed a new system that implements demand response without big changes to existing infrastructure. It does not require collecting and processing large amounts of data. It ties electricity price to grid frequency. The proposed work has some assumptions like fixed inputs issue and equality issue. To deal with these two, the system is investigated for different design points and decision trees are applied to results. To deal with large inputs, inputs are aggregated to features. It is found that fast adaptation generally improves stability. For delays bigger than 8s system is found to be always unstable. Paper finds in a stable grid consumer may have a reaction time longer than 8s as long as another consumer is reacting fast and the average reaction time is moderate. The slow-reacting consumer is found to be advantageous in this situation. The trade-off between avoiding the rebound effect with small reaction times and increasing the basin with high values of reaction time is shown. High averaging times were found to have a positive effect on stability. Power is found to have little no effect on stability. However, there is a space for more generalization. Also, the paper archives 80% accuracy, which is fine for a general view but not good enough for finding pure stability regions. Extension of this analysis to large grids with more than 10 users is also not straightforward.

Networked Control System (NCS) suffers from packet dropout, delays and packet disorder. These are degrading

to performance. In power system literature transmission of signals are considered ideal, lossless and relay-free. Singh et al. [2] analyzed the effects of these in networked controlled smart grid (NCSG). Both UDP and TCP are considered for network. Although UDP-like protocol results in a sub-optimal solution, it is preferred over a TCP-like protocol as it may be extremely difficult to both analyze and implement a TCP-like control scheme [5]. Packets are considered lost if a delay is more than the sampling interval. Loop is considered to be stabilized and damped when no packet loss occurs. Packet loss results in loss of output measurements and therefore lower controller estimates. A case study was made with 16 machine 68 bus model, which is reduced-order equivalent of interconnected New England test system (NETS) and New York power system (NYPS). A sampling period of 0.1 s is chosen. Performance of NCS with minimal packet dropout even with marginal packet delivery probability (MPDP) is found to comparable to classical control of an ideal system. NCS is found better when both systems are in ideal condition. NCS can damp oscillations in a smaller amount of time even when the controller effort required by NCS is decreased by 22%. NCS is better than classical controller at realistic expected packet delivery quality. Higher sampling rates result in a better-needed packet delivery rate. It is found that NCSG is stable a delivery probability of 0.85. Research shows that NCSG has good potential to deliver oscillatory stability margin for a smart grid.

Ayar et al. [6] proposed a distributed nonlinear robust controller. Smart grids raise cyber-attack risks and inherent psychical limits. Phasor Measurement Unit (PMU) raises stability and security by giving real-time system state but also causes big data volume and this can cause latency. With the displacement of high inertia generation with lower inertia generation more robustness is needed. A nonlinear distributed controller robust to big input and communication delays that do not need exact model knowledge is presented. All 10 generators were found to converge in about 3.5 s or earlier with an input delay of 100 ms and a response time of 20 ms. Control is evaluated for practical limits and compared to Parametric Feedback Linearization (PFL) controller. Stabilization time decreases as DSS Max power ratio increases until it hits more than 10%. PFL controller is not equally robust to latency, even though stabilization times are similar when the latency is 50 ms it is found that in latency over 50 ms PFL cannot stabilize the system. Both can stabilize if distributed storage system (DSS) max power ratio is greater than 5. If latency is over 100 ms PFL cannot stabilize the generators, the proposed controller can do it for all systems under 800 ms. The proposed controller outperforms PFL under larger latencies. The proposed controller can compensate up to 1 s latency plus 20 ms control input delay.

The event leads to disruption includes DoS, false data injection, switching attacks and physical faults. Farraj et al.

[7] focused on reactive approaches for DoS and switching attacks to boost resilience against. PFL uses EES to give or take energy from storage. Other relevant issues like voltage and signal stability and wide area control are left out. The proposed controller is delay adaptive and adapts to comm latency and state of the grid. New England (NE) 39 bus and WECC 9 bus are used for the test. Delays in this paper are the total delay between sensors and controllers. PFL controller utilizes system info to synchronize generator more aggressively in unstable state. A cyber-psychical PFL controller is designed to have a gain scheduling design. DFL controller for agents with no local sensors is also proposed. Delay the in-adaptive controller found to stabilize when the latency is below 150 ms however Cyber-psychical FPL controller is found to stabilize when the latency is below 225 ms. Also proposed controller has a great power gain over in-adaptive controller, especially for high comm latency. With DFL controller, different generators are found to stabilize quickly but generators without local sensors are found to take more time. Including other types of cyber-attacks, researching cyber aware wide-area controller schemes and researching other types of controller schemes are suggested as further directions.

Bejestani et al. [8] proposed hierarchical transactive control architecture. Supply and demand balance in power grids are maintained using hierarchical control schemes with multiple time scales. With an unit level primary level, area-level secondary level and tertiary level where Economic Dispatch occurs. They addressed regulation response Demand Response (DR) and price response DR for a primary controller. They proposed a dynamic dispatch system based on GenCo, ConCo and ISO exchange info through low latency communication network. Dynamic Market Mechanism that uses state space, state-dependent payoff functions, actions and d-step ahead prediction is also introduced for the market. It is found that fewer reserves are need using transactive control while wind power fluctuations happen, mainly due to coordination between levels and one step ahead prediction in the Dynamic Market Mechanism (DMM). Proposed controller results in larger following wind-intermittency and less deployment of the reserve with lower reserve cost. Proposed controller results in fewer generation costs and reserve cost reduction. It is found that implementing it can improve social welfare from 97.8 \$/h to 134.2 \$/h. Solution time that is needed for iterations are found to be small enough for practical implementations. The solution was tested in 4, 30 and 118 bus systems. For a data rate of 1 mbs, 10^7 iterations are 1.5 communication time which is fine for a real-time market for many of the ISOs. One of the generators was disconnected. The proposed system compensated for it. No restoration was provided without a proposed controller.

Post disturbance Transient Stability Assessment (TSA) can be applied in real-time. To be considered real-time

TSA needs to find results before the next measurement data comes. ML algorithms are used in real-time TSA system. Used methods can be categorized as offline-online or they can only use post disturbance data for stability. These are dependent to mathematical model of system. High complexity and execution times limit emergency control. Time delays adds noise in Wide Area Measurement Systems (WAMS). Results are probabilistic which limits practical repeatability. For solving these Sliding Window Based TSA (SW-TSA) is suggested for large grids [9]. The power system is complex so nonlinear MOR is used. For Model Order Reduction (MOR), Proper Orthogonal Decomposition (POD) is used and Randomized SVD is chosen for its simplicity and speed, as POD operator. SWA-TSA in order does: (i) measurement and construct snapshot matrix, (ii) POD, (iii) estimate low dimensional model, (iv) predict for high dimensional states and do TSA, (v) do emergency action if needed. For SWA-TSA two sliding windows are used; for POD SSW is used, for predicting PSW is used. The method also finds unstable machines. A case study is done on two areas, 4 machine 11 bus system and proposed (AR) method is compared to state space, prony and polynomial models. The space state method was found to have a better prediction rate but AR was found to be simpler and faster. NE 10 machine system is used and TSI-COI and TSI-RRA are tested for finding system stability. Lower instability times are calculated on TSI-COI compared to TSI-RRA. Next a country-wide system is tested and only TSI-COI is used. Instability time found is less than, 5 ms max for real time systems. The proposed system was also found to predict multi-swing stability. The proposed algorithm is composed of three sub-systems and among these SVD has the biggest cost, for this cost scales linearly with a number of machines and for a large number of machines GPU's can be used. SWA-TSA is also shown to predict stability and stable machines in case of missing data. Since SWA-TSA has independent iterations needed parameters can change while the system is running. Prediction windows also need to be at least 0.2 s.

Schäfer et al. [10] proposed DSGC for balancing supply and demands using grid frequency information. The key idea is frequency provides all information is needed since frequency increases in times of excess generations and decreases at underproduction times. A system where consumers' price of electricity is a direct function of grid frequency is proposed and it is shown that for this system certain delay values are risky. Paper proposes averaging frequency measured to solve this. Proposed DSGC realizes DR and analyzes its economic effects and also calculates prices based on angular frequency deviation but since measuring it takes time paper uses time-averaged frequency deviation. Paper assumes supply and demand show the same characteristics on scales given. The proposed system has several economic advantages like: it can change the demand to

lower-priced times, DR allows global costs of the system, DR may improve stability by avoiding big peaks of power usage and may improve market performance by reducing deviation. First, DR that is faster than the grid is tested and found to always lower return time after perturbation. A more realistic scenario where control is slower than the grid is also tested. Cases where rebound effects occur is found but similar effects can occur all DR systems. The proposed system is tested in IEEE 9 bus test grid. It is found that two systems may become interdependent when DSGC acts similar time scales. For this issue paper suggests linearizing supply and demand curves. This system will either damp or amplify depending on the phase. Next, grid dynamics after the large perturbation is considered. Depending on delay stability found to vary. For 0.8 fixed point was found to be unstable but for 5 s it has almost perfect stability. Paper shows, for real-world system, delay in adaptation possesses a stability risk but averaging the signal over time ensures stable operation. Nodes can adapt to generation changes in all time scales slower than averaging time. The paper shows either delayed adaptation must be avoided or for a long enough period of T averaging must be done.

Climate change is forcing us to renewable sources but normal power plants still dominate power grids and in these, transmission lines connect in a star like topology locally, with plants in the center and consumers at edges. In [10], it was found that averaging improves stability in DSGC but it was only tested in small grids. Schäfer et al. [3] analyzed impact of topologies in stability both using linear stability and determining basin value. How grid stability changes when generation becomes decentralized is also shown. Proposed DSGC is based on DR and similarly to [11] uses oscillator model to model. Consumers shown with a linear power price relation even though consumers act complexly. In such system, [10] showed risks arise to solve this averaging was done. This paper chooses homogeneous averaging times and treats it as control parameter, and similar delays for all machines is chosen. Dynamic stability around the steady-state operation, driven by fixed point is analyzed to study role of parameters. For larger perturbations, estimating basin volume using a Monte Carlo method was proposed with first simulating perturbing a producer node and then simulating perturbing a consumer node. This simulations found to be costly but needed. Multiple nodes may be perturbed at same time but resulting phase space becomes infeasible to sample. Proposed analysis is tested on a four-node star motif with one producer at center. Intermediate delays are shown to benefit stability. Star topologies found to heavily depend on delay and averaging time. Without averaging, value of delays where fixed points are linearly unstable is found. It is also found that with high enough averaging times and for delays under 7 these disappear. For larger delays than critical delay, rebound effect occurs and system always

gets destabilized. But it is also found that critical delay only exists if the price adaptation is larger than intrinsic damping of the system. Approximation for this critical value is found. These findings are supported by basin volume analysis. It is found that for delay value of 4 close to perfect stability with basin volume of 1 for an averaging time of both 1s and 2s achieved. Basin volume also reveals consumer node disturbances are less likely destabilize the system compared to producer. Switching centralized to decentralized production improves linear stability. For critical coupling of $4/s^2$ central production can't be stabilized. More connected topology like lattice topology is found to allow central production at this value and even $2/s^2$ with large averaging times and perform overall better. Overall averaging time of 4 is found to stabilize DSGC for all delays. Decentralized production is found to make advantage of smaller distances to consumers and allow smaller averaging times. To the our knowledge, there is no work for smart grid stability prediction based on deep learning-based techniques in the literature.

Abedinia et al. [11] studied on energy procurement problem of large industrial consumers and proposed a hybrid robust-stochastic approach. Uncertainty arising from the load was modeled via robust optimization. Saeedi et al. [12] studied on optimal scheduling of electrical power consumption in the multi-chiller system. Robust optimization based results were compared with a deterministic method. Gao et al. [13] proposed a new prediction model for price forecast using ANN, SVM and RBFNN. A modified ordered weighted average was used for improving the proposed fusion algorithm. Gadimi et al. [14] proposed a two-stage forecast engine consists of RNN and ENN. An effective prediction model for training the hybrid forecast engine was presented. Khodaei et al. [15] proposed heat and power hub model to supply power and heat demands. A fuzzy decision-making approach was provided to select the trade-off solution from the Pareto solutions. Begal et al. [16] obtained a risk-neutral strategy with a deterministic approach without uncertainty modeling.

Materials and Methods

In this section, the proposed optimized deep models for predicting smart grid stability are given. Fig. 1 shows a flowchart-based illustrative figure that demonstrates that how the provided strategy would work.

Dataset Description

The dataset consists of results taken from grid stability simulations done on a 4-node network with star topology (see Fig. 2), as described in [10] and has a synthetic nature. It has 12 primary predictive features and two dependent variables.

Fig. 1 Flowchart for the proposed strategy

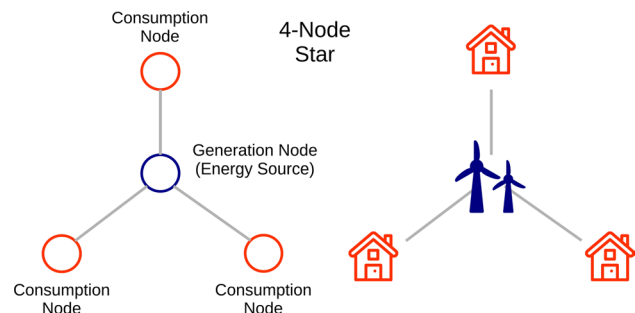
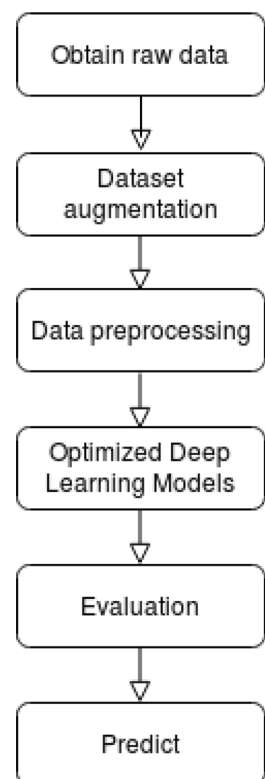


Fig. 2 Four-node star architecture

It has 10,000 samples. Since the grid used in simulations is symmetric we can augment the original dataset 3! (6) times. The augmented dataset consists of 60,000 samples. Table 1 shows the first five samples of the augmented dataset.

Predictive features:

- 'tau1' to 'tau4': the reaction time of each network participant, a real value within the range 0.5–10 ('tau1' corresponds to the supplier node, 'tau2' to 'tau4' to the consumer nodes);
- 'p1' to 'p4': nominal power produced (positive) or consumed (negative) by each network participant, a real value within the range -2.0 to -0.5 for consumers ('p2' to 'p4'). As the total power consumed equals the total power generated, $p1$ (supplier node) = $-(p2 + p3 + p4)$;

Table 1 The first five samples of used dataset

	tau1	tau2	tau3	tau4	p1	p2	p3	p4	g1	g2	g3	g4	stab	stabf
46496	4.475683	4.836149	9.864130	6.348654	4.711574	-1.783794	-1.324899	-1.602881	0.940242	0.593144	0.600571	0.415185	0.058197	0
44408	7.142503	6.863533	3.417915	8.086431	2.248925	-1.012613	-0.618024	-0.618288	0.163990	0.823578	0.280829	0.698357	0.048240	0
2854	3.625925	3.303132	6.691957	9.852377	4.277447	-1.212106	-1.938179	-1.127162	0.369874	0.093810	0.296905	0.332527	-0.025830	1
19176	8.315507	8.019450	5.581036	1.269828	4.002283	-0.909118	-1.847469	-1.245697	0.125373	0.823954	0.475396	0.324136	0.017061	0
35380	6.582167	8.979464	1.231033	8.354518	3.355258	-0.646911	-1.799268	-0.909079	0.652189	0.355355	0.904233	0.792863	0.028298	0

- 'g1' to 'g4': price elasticity coefficient for each network participant, a real value within the range 0.05 to 1.00 ('g1' corresponds to the supplier node, 'g2' to 'g4' to the consumer nodes; 'g' stands for 'gamma');

Dependent variables:

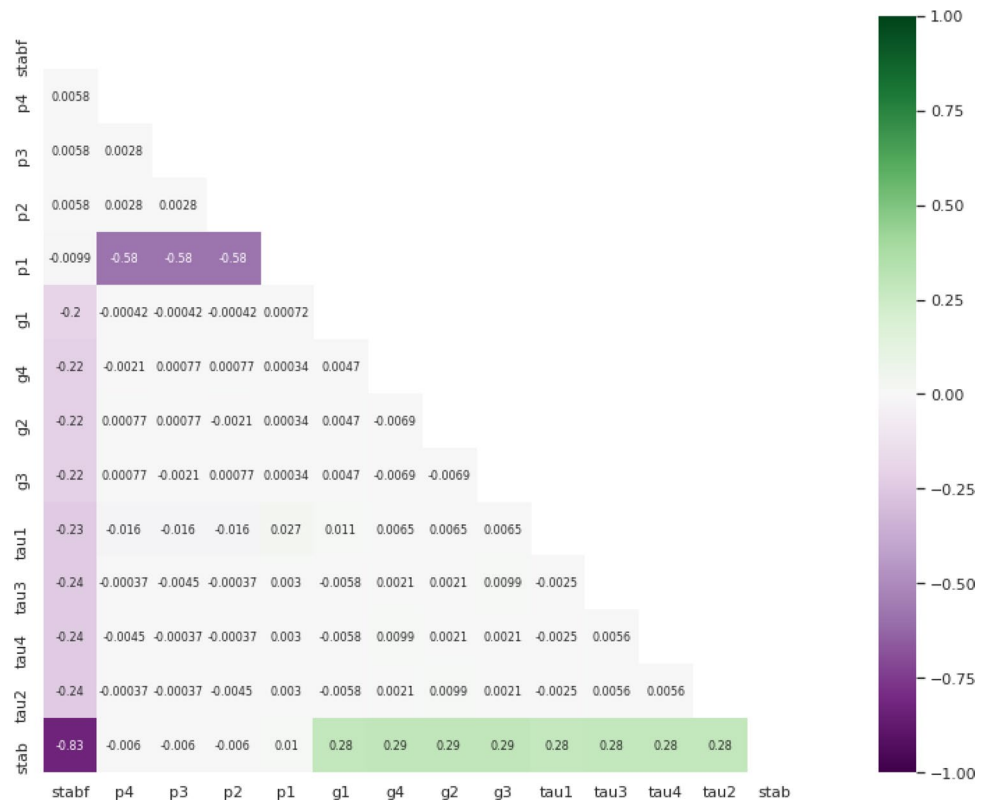
- 'stab': the maximum real part of the characteristic differential equation root (if positive, the system is linearly unstable; if negative, linearly stable);
- 'stabf': a categorical (binary) label ('stable' or 'unstable').

'stabf' field is 0 if 'stab' value is higher than 0, and 1 otherwise. This means there is a direct relationship between these fields. We choose to drop "stab" field and use 'stabf' as a only dependent variable. Since features come from simulations compared to real-world measurements, the dataset does not have any missing values. This, coupled with the dataset only consisting of numerical values allows us to skip data pre-processing steps and jump to machine modeling. For all features, distribution patterns and relationship with the chosen dependent variable is charted. 'p1' is an absolute sum of 'p1', 'p2', 'p3', 'p4'. Since the dataset values are from simulation; all features have fixed ranges and distributions are mostly uniform across to board with 'p1' being the exception, which follows a normal distribution with the skew value of -0.013.

It is important to verify the correlation between each numerical feature and the dependent variable, as well as the correlation among numerical features leading to potential undesired co-linearity. Figure 3 shows the correlation between the chosen dependent variable and numerical features. This is required considering correlation between numerical features may lead to co-linearity. We observe a significant correlation (-0.83) between dropped and chosen dependent value, this supports our decision to drop 'stab'. We, also, see the above-average correlation between 'p1' and its sum components. In the end, we don't consider this grave enough to remove 'p1'.

Optimized Deep Learning Models

According to the current situation most of the researchers have concentrated their studies on DL because DL has been treated as one of the emerging areas for feature extraction and handling a huge amount of data where machine learning methods fail. For electricity data analysis industries were adopted novel methods like machine learning, fuzzy logic, data mining, artificial neural network (ANN), support vector machine (SVM) and genetic algorithm, etc. to get better outcomes for estimating the exact electricity demand and also used these methods to forecast both energy production

Fig. 3 Correlation matrix for dataset attributes

and consumption. Among all methods deep learning plays a vital role in smart grids applications. In this study, different optimized DL models are used to predict SG stability. Details are given in the next section.

Neural network researchers have long realized that the learning rate is reliably one of the most difficult to set hyperparameters because it significantly affects model performance. The cost is often highly sensitive to some directions in parameter space and insensitive to others. The directions of sensitivity are somewhat axis aligned, it can make sense to use a separate learning rate for each parameter and automatically adapt these learning rates throughout the course of learning. A number of incremental (or mini batch-based) methods have been introduced that adapt the learning rates of model parameters such as Adam and Nadam optimizer.

Experimental Results

Experimental Setup

Along with traditional libraries imported for tensor manipulation, mathematical operations and graphics development, three scikit-learn modules (StandardScaler as a scaler, confusion_matrix as the model performance metric of choice and KFold as the cross-validation engine) and two Keras deep learning objects (Sequential and Dense) are used in

this paper [17]. The original 10,000 observations) and augmented dataset (60,000 observations) are imported. The dependent variable is map encoded ('stable' replaced with 1, 'unstable' with 0). At last, the 60,000 observations are shuffled.

As anticipated, the features dataset will contain all 12 original predictive features, while the labeled dataset will contain only 'stabf' ('stab' is dropped here). In addition, as the dataset has already been shuffled, the training set will receive the first 54,000 observations, while the testing set will accommodate the last 6000. Even considering that the

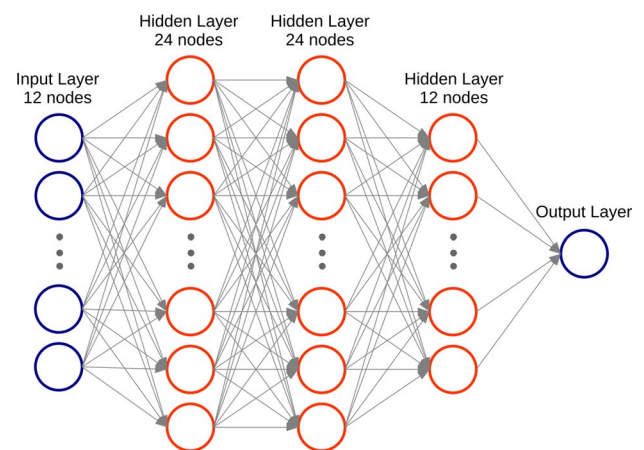
**Fig. 4** 24x24x12x1 architecture

Table 2 Performance results for augmented dataset

Architecture	Optimizer	Folds	Epochs	Confusion matrix		Accuracy (%)	Precision (%)	F1-Score (%)
24-24-12-6-1	Adam	10	10	3746	51	97.55	98.65	98.07
				96	2107			
24-24-12-6-1	Adam	10	20	3804	36	97.60	99.06	98.14
				108	2052			
24-24-12-6-1	Adam	10	50	3762	36	97.68	99.05	98.18
				103	2099			
24-48-24-12-1	Adam	10	10	3802	39	98.13	98.98	98.44
				73	2086			
24-48-24-12-1	Adam	10	20	3792	33	98.22	99.13	98.54
				74	2101			
24-48-24-12-1	Adam	10	50	3770	57	98.02	98.51	98.60
				62	2111			
24-48-48-12-1	Adam	10	10	3836	59	98.08	98.48	98.52
				56	2049			
24-48-48-12-1	Adam	10	20	3770	59	98.33	98.45	98.69
				41	2130			
24-48-48-12-1	Adam	10	50	3840	41	98.25	98.94	98.65
				64	2055			
24-48-48-12-1	SGD(Learning=0.01,Momentum=0.9)	10	10	3784	74	98.25	98.06	98.61
				31	2147			
24-48-48-12-1	SGD(Learning=0.01,Momentum=0.9)	10	20	3821	25	98.23	99.35	98.63
				81	2073			
24-48-48-12-1	SGD(Learning=0.01,Momentum=0.9)	10	50	3721	74	97.88	98.05	98.32
				53	2152			
288-288-24-12-1	Adam	10	10	3800	24	99.23	99.37	99.39
				22	2154			
288-288-24-12-1	Adam	10	20	3763	23	98.68	99.39	98.96
				56	2158			
288-288-24-12-1	Adam	10	50	3790	22	99.35	99.42	99.48
				17	2171			
288-288-24-12-1	Nadam	10	10	3862	37	99.00	99.05	99.22
				23	2068			
288-288-24-12-1	Nadam	10	20	3763	31	98.83	99.18	99.07
				39	2167			
288-288-24-12-1	Nadam	10	50	3791	15	99.62	99.63	99.69
				8	2186			
288-288-24-12-1	Nadam	20	10	3784	31	99.07	99.18	99.26
				25	2160			
288-288-24-12-1	Nadam	20	25	3801	14	99.60	99.60	99.68
				10	2175			
144-144-12-1	Nadam	10	10	3715	71	98.30	98.12	98.64
				3	2183			
144-144-12-1	Nadam	10	20	3742	52	98.70	98.62	98.96
				26	2189			
144-144-12-1	Nadam	10	50	3791	28	98.20	99.26	99.25
				29	2152			
216-216-216-12-1	Nadam	10	10	3867	28	99.13	99.28	99.33
				24	2081			

Table 2 (continued)

Architecture	Optimizer	Folds	Epochs	Confusion matrix		Accuracy (%)	Precision (%)	F1-Score (%)
216-216-216-12-1	Nadam	10	20	3794	23	99.15	99.39	99.33
				28	2155			
216-216-216-12-1	Nadam	10	50	3841	23	99.28	99.40	99.32
				29	2116			

Bold values indicate the best results

dataset is large enough and well behaved, the percentage of 'stable' and 'unstable' observations is computed for both training and testing sets, just to make sure that the original dataset distribution is maintained after the split—which proved to be the case.

Model Definition and Fitting

In preparation for machine learning, scaling is performed based on (fitted to) the training set and applied (with the 'transform' method) to both training and testing sets. Multiple models are used for comparison. One of them is depicted in Fig. 4. It reflects a sequential structure with: one input layer (12 input nodes), three hidden layers (24, 24 and 12 nodes, respectively), one single-node output layer. We used "relu" activation function in hidden layers and "sigmoid" activation function in output layers [18]. For real number features and binary outputs like ours these are the most accepted choices in literature. The choices of binary cross-entropy for our loss function [19] and accuracy as our performance metric follows a similar logic.

Along with different models we also used different optimizers. "Adam" optimizer [20] for its fast learning times, adaptive learning rates and overall great results and Stochastic Gradient Descent (SGD) with momentum [21] for its ability to generalize better in some cases are the most used optimizers in literature. Along these two, we also picked "Nadam" optimizer (Which combines NAG with "Adam" optimizer) to see if Nesterov Gradient [22] improves the result on our test case.

We use a cross-validation-based fitting because our dataset is uniformly distributed. For our cross-validation engine KFold is chosen. We use K values of 10 and 20. For the K value of 10, epoch numbers of 10, 20 and 50 are chosen. For K value of 20 and for same values of epoch number run time is roughly doubled so we choose epoch numbers of 10 and 25 instead. Performance of different models, optimizers, fold values and epoch values are discussed in the next section.

Performance Metrics

Models are tested both on augmented and original datasets. No shuffling is done for both data sets so for all model/optimizer combinations the same datasets are used. This eliminates the variable of shuffling for results. For output values threshold of 0.5 is chosen. Any value above 0.5 is classified as "stable" and any value below is classified as "unstable". In this study, we give a performance of DL architectures in terms of accuracy, precision, and f1-score metrics derived from the confusion matrix as the following equations:

$$\text{accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

$$\text{precision} = \frac{TP}{TP + FP} \quad (2)$$

$$\text{sensitivity (recall)} = \frac{TP}{TP + FN} \quad (3)$$

$$f1 - \text{score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (4)$$

Performance Results

Tables 2 and 3 show performance results for augmented and original datasets, respectively. The best performance results are in bold in the tables.

As expected, the use of augmented data set with 6 times of samples compared to the original dataset significantly improved the results. We can also see similarly improved performance gains in increasing epoch number. This supports the notion that the more the model has been exposed to the training dataset, the better prediction it gives. Increasing the fold number by twice and halving the epoch number gives similar results for the original dataset but as we can see it gives worse results compared otherwise.

Table 3 Performance results for original dataset

Architecture	Optimizer	Folds	Epochs	Confusion Matrix		Accuracy (%)	Precision (%)	F1-Score (%)
24-24-12-6-1	Adam	10	10	620	19	94.55	97.02	95.75
				36	325			
24-24-12-6-1	Adam	10	20	597	32	94.92	94.91	95.90
				19	352			
24-24-12-6-1	Adam	10	50	631	10	95.90	98.43	96.85
				31	328			
24-48-24-12-1	Adam	10	10	615	29	95.38	95.49	96.31
				18	338			
24-48-24-12-1	Adam	10	20	607	33	94.21	94.84	95.44
				25	335			
24-48-24-12-1	Adam	10	50	610	12	96.30	98.07	97.05
				25	353			
24-48-48-12-1	Adam	10	10	591	32	92.40	94.86	93.95
				44	332			
24-48-48-12-1	Adam	10	20	642	16	95.35	97.56	96.46
				31	311			
24-48-48-12-1	Adam	10	50	621	24	95.42	96.27	96.42
				22	333			
24-48-48-12-1	SGD(Learning=0.01,Momentum=0.9)	10	10	621	30	94.24	95.39	95.53
				28	321			
24-48-48-12-1	SGD(Learning=0.01,Momentum=0.9)	10	20	603	19	95.65	96.94	96.48
				25	353			
24-48-48-12-1	SGD(Learning=0.01,Momentum=0.9)	10	50	599	19	95.81	96.92	96.61
				23	359			
288-288-24-12-1	Adam	10	10	612	22	96.08	96.52	96.83
				18	348			
288-288-24-12-1	Adam	10	20	627	29	96.60	95.57	96.68
				14	339			
288-288-24-12-1	Adam	10	50	602	18	97.14	97.09	97.64
				11	369			
288-288-24-12-1	Nadam	10	10	627	14	96.93	97.81	97.58
				17	342			
288-288-24-12-1	Nadam	10	20	623	13	96.50	97.95	97.26
				22	342			
288-288-24-12-1	Nadam	10	50	627	11	97.52	98.67	98.24
				14	348			
288-288-24-12-1	Nadam	20	10	614	19	96.77	96.99	97.38
				14	353			
288-288-24-12-1	Nadam	20	25	621	15	96.72	97.64	97.41
				18	346			
144-144-12-1	Nadam	10	10	626	29	95.95	95.57	96.82
				12	333			
144-144-12-1	Nadam	10	20	640	18	96.81	97.26	97.56
				14	328			
144-144-12-1	Nadam	10	50	611	16	97.10	97.44	97.68
				13	360			
216-216-216-12-1	Nadam	10	10	609	25	96.40	96.05	97.12
				11	355			

Table 3 (continued)

Architecture	Optimizer	Folds	Epochs	Confusion Matrix		Accuracy (%)	Precision (%)	F1-Score (%)
216-216-216-12-1	Nadam	10	20	641	18	96.94	97.26	97.63
				13	328			
216-216-216-12-1	Nadam	10	50	632	12	97.51	98.13	98.13
				12	348			

Bold values indicate the best results

Table 4 Comparison of prediction accuracy using different techniques

Prediction method	Accuracy (%)	Used system
XGBoost [23]	97.8	IEEE 39-bus test system
Bayesian rate (BR) [24]	91.6	IEEE 39-bus test system
DT [25]	90.3	IEEE 39-bus test system
CNN [26]	89.22	IEEE 118-bus and IEEE 145-bus systems
Long Short Term Memory (LSTM) [27]	99.98	IEEE 39-bus test system and 162 and 145 bus systems
FNN-based transient stability predictor [28]	99.2	IEEE 39-bus test system
CART [4]	80.0	4-node star architecture
Proposed	99.62	4-node star architecture

Inspecting models, we can see the model with the best performance and the most optimal model is “288-288-24-12-1”, which mean it has 288 nodes in its first hidden layer, 288 nodes in its second hidden layer, 24 nodes in its third hidden layer and 12 nodes in its fourth hidden layer. For this and other models “nadam” optimizer gives slightly better results while being slightly slower to run but such a small difference in run time can be considered irrelevant. The codes are available at: <https://github.com/pcbreviglieri/data-science-smart-grid-stability>

Formal Comparison with State-of-the-Art Works

Grid stability detection is a very significant topic and plenty of work have been carried out in this domain for a long. Table 4 shows a summary of the results. It includes different simulation systems. Our optimized DL models show that stability prediction has promising results in terms of local (linear) stability analysis. Linear stability analysis explores dynamical stability around the steady-state operation of the grid.

Conclusion

The advent of renewable energy comes with its changes and challenges. Smart grid take supply and demand information from the power grid and calculate price information. DSGCs only require frequency information, which

is easily accessible anywhere in the system. Previously researched [3, 4] ways gives us a model to predict grid stability but comes with significant simplifications. These simplifications cause issues.

In this paper, we proposed using deep learning for stability prediction to solve these issues. We use dataset taken from simulations and test multiple deep learning datasets with them. As well as numerous different deep learning architectures we also compare optimizers for deep learning and tinker with different fold/epoch values. Models that have upwards of 99.6% accuracy are found. For the most optimal model, “nadam” was found to be the optimizer choice. Our research shows that deep learning models give new insights into the simulated system.

It must be noted that input parameters utilized in the original DSGC simulations fall within predetermined ranges. As a follow-up step in the validation of this learning machine, it would be interesting to assess its performance using a new test set with observations obtained from simulations with input parameter values residing in other alternative ranges.

It is important to note that the input parameters used in DSGC simulation were in predetermined ranges. In the future, using a dataset from DSGC simulations where input parameters are in alternative ranges can be done.

Funding The authors received no financial support for the research, authorship, and/or publication of this article.

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

Ethical Approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

- Eltigani D, Masri S. Challenges of integrating renewable energy sources to smart grids: a review. *Renew Sustain Energy Rev*. 2015;52:770–80.
- Singh AK, et al. Stability analysis of networked control in smart grids. *IEEE Trans Smart Grid*. 2014;6(1):381–90.
- Schäfer B, et al. Taming instabilities in power grid networks by decentralized control. *Eur Phys J Spec Top*. 2016;225(3):569–82.
- Arzamasov V, et al. Towards concise models of grid stability. In: 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (Smart-GridComm), IEEE, 2018, pp. 1–6.
- Sinopoli B, et al. Optimal linear lqg control over lossy networks without packet acknowledgment. In: *Proceedings of the 45th IEEE Conference on Decision and Control*, IEEE, 2006, pp. 392–397.
- Ayar M, et al. A distributed control approach for enhancing smart grid transient stability and resilience. *IEEE Trans Smart Grid*. 2017;8(6):3035–44.
- Farraj A, et al. A cyber-physical control framework for transient stability in smart grids. *IEEE Trans Smart Grid*. 2016;9(2):1205–15.
- Bejestani AK, et al. A hierarchical transactive control architecture for renewables integration in smart grids: analytical modeling and stability. *IEEE Trans Smart Grid*. 2014;5(4):2054–65.
- Shamisa A, et al. Sliding-window-based real-time model order reduction for stability prediction in smart grid. *IEEE Trans Power Syst*. 2018;34(1):326–37.
- Schäfer B, et al. Decentral smart grid control. *New J Phys*. 2015;17(1):015002.
- Abedinia O, Zareinejad M, Doranehgard MH, Fathi G, Ghadimi N. Optimal offering and bidding strategies of renewable energy based large consumer using a novel hybrid robust-stochastic approach. *J Clean Prod*. 2019;215:878–89.
- Saeedi M, Moradi M, Hosseini M, Emamifar A, Ghadimi N. Robust optimization based optimal chiller loading under cooling demand uncertainty. *Appl Therm Eng*. 2019;148:1081–91.
- Gao W, Darvishan A, Toghiani M, Mohammadi M, Abedinia O, Ghadimi N. Different states of multi-block based forecast engine for price and load prediction. *Int J Electr Power Energy Syst*. 2019;104:423–35.
- Ghadimi N, Akbarimajd A, Shayeghi H, Abedinia O. Two stage forecast engine with feature selection technique and improved meta-heuristic algorithm for electricity load forecasting. *Energy*. 2018;161:130–42.
- Khodaei H, Hajiali M, Darvishan A, Sepehr M, Ghadimi N. Fuzzy-based heat and power hub models for cost-emission operation of an industrial consumer using compromise programming. *Appl Therm Eng*. 2018;137:395–405.
- Bagal HA, Soltanabad YN, Dadjuo M, Wakil K, Ghadimi N. Risk-assessment of photovoltaic-wind-battery-grid based large industrial consumer using information gap decision theory. *Solar Energy*. 2018;169:343–52.
- Pedregosa F, et al. Scikit-learn: machine learning in python. *J Mach Learn Res*. 2011;12:2825–30.
- Agostinelli F, et al. Learning activation functions to improve deep neural networks. *arXiv preprint:1412.6830*, 2014.
- Ho Y, Wookey S. The real-world-weight cross-entropy loss function: modeling the costs of mislabeling. *IEEE Access*. 2019;8:4806–13.
- Zhang Z. Improved adam optimizer for deep neural networks. In: 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), IEEE, 2018, pp. 1–2.
- Cutkosky A, Orabona F. Momentum-based variance reduction in non-convex sgd. In: *Advances in Neural Information Processing Systems*, 2019, pp. 15236–15245.
- Jakovetić D, et al. Fast distributed gradient methods. *IEEE Trans Autom Control*. 2014;59(5):1131–46.
- Chen M, Liu Q, Chen S, Liu Y, Zhang C-H, Liu R. Xgboost-based algorithm interpretation and application on post-fault transient stability status prediction of power system. *IEEE Access*. 2019;7:13149–58.
- Zare H, Alinejad-Beromi Y, Yaghobi H. Intelligent prediction of out-of-step condition on synchronous generators because of transient instability crisis. *Int Trans Electr Energy Syst*. 2019;29(1):e2686.
- Mohammad R, Aghamohammadi F, Morteza A. Dt based intelligent predictor for out of step condition of generator by using pmu data. *Int J Electr Power Energy Syst*. 2018;99:95–106.
- Gupta A, Gurralla G, Sastry PS. An online power system stability monitoring system using convolutional neural networks. *IEEE Trans Power Syst*. 2018;34(2):864–72.
- James JQ, Hill David J, Lam Albert YS, Gu J, Li VOK. Intelligent time-adaptive transient stability assessment system. *IEEE Trans Power Syst*. 2017;33(1):1049–58.
- Darbandi F, Jafari A, Karimipour H, Dehghantanha A, Derakhshan F, Choo KKR. Real-time stability assessment in smart cyber-physical grids: a deep learning approach. *IET Smart Grid*. 2020;3(4):454–61.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.