AUTOMATIC DETECTION OF DISTRIBUTED SOLAR GENERATION BASED ON EXOGENOUS INFORMATION

Aleksandr Petrusev, Rémy Rigo-Mariani, Vincent Debusschere, Patrick Reignier, Nouredine Hadjsaid

AUTOMATIC DETECTION OF DISTRIBUTED SOLAR GENERATION BASED ON EXOGENOUS INFORMATION

Aleksandr Petrusev, Rémy Rigo-Mariani, Vincent Debusschere, Patrick Reignier, Nouredine Hadjsaid

Univ. Grenoble Alpes, CNRS, Grenoble INP, F-38000, Grenoble, France
aleksandr.petrusev@grenoble-inp.fr

Keywords: ARTIFICIAL INTELLIGENCE, AUTOMATIC PV DETECTION, NEURAL NETWORK.

Abstract

In recent years, the importance of PV generation data for distribution system operations has increased. However, there are still a lot of behind-the-meter solar installations that are not registered by the system operator and are not monitored. This "hidden" generation, therefore, increases the difficulty to operate securely and efficiently the distribution grid. This paper introduces a tool for the automatic detection of such "hidden" behind-the-meter solar generation. It is designed to discriminate the nodes with and without PV generation and is aimed at a high accuracy. The tool consists of a neural network coupled with an analytical classification algorithm, which considers an exogenous information (i.e. node consumption and temperature data). Open-access data about consumption and solar radiation were used to simulate the electrical grid and validate the proposed approach. The implemented solution was tested across all the nodes of the grid and its sensitivity has been analysed with regard to the level of PV penetration and period of observation. The tool is able to recognize the nodes with a new PV installation with an accuracy of up to 100%, depending on the exogenous conditions.

1 Introduction

Distributed renewable energy, and particularly photovoltaic (PV), showed rapid growth over the past two decades. For this reason, information on PV generation becomes one of the key aspects to perform operations such as voltage management, reconfiguration or state estimation in distribution systems. However, a large number of small household-owned PV plants do not necessarily have connection agreement with the system operator and are therefore not monitored. That generation is then "hidden" for the system operator. It incurs additional uncertainty in the net charge measurement and forecasting, which may affect the quality of the distribution grid operation. Thus, a relevant contribution to improve electrical grid management performance is to increase the system observability with a tool that detects nodes with PV production, based on net charge data. There are several proposed approaches for detecting PV generation in distribution grids, but they often require specific data that the system operator may not have access to, such as solar radiation data.

One of the approaches, which identifies customers with PV generation by using data about net energy consumption from smart meters, is introduced in [1]. The authors propose a dimensionality reduction method combined with a classification algorithm, more effective than K-means, to reduce the amount of data needed to accurately identify solar prosumers to a single data point (by taking the minimum, average, and maximum consumption for the entire year). The classification method is based on agglomerative clustering and self-organizing maps (an artificial neural network-based clustering technique). One limitation of the method is that the clustering of consumers into two groups (with and without PV) is carried out based on whether the peak consumption of a consumer is below the average of all customers, without taking into account data on the historical consumption of this consumer in previous years. Thus, the performances and accuracy are strongly affected by the diversity of the considered consumption load profiles.

Another algorithm proposes the automatic detection of PV panels by using very high-resolution color satellite imagery (0.3 meters per pixel) [2]. A Random Forest classification machine learning technique detects the presence of PV on an image. The need to use high-resolution satellite images along with labeled training data is the main disadvantage of this algorithm.

An approach for solar prosumer identification, which uses change-point to detect abnormal energy consumption behaviors (such as not monitored PV generation), is presented in [3]. Various abnormalities can cause change-points in customer load. Therefore, the presence of the not monitored PV generation is further verified through a statistical inference known as permutation test with Spearman's rank coefficient. Nevertheless, this approach is unable to detect PV installations until after the rolling window length is completed (up to 14 days lengths were evaluated), and additionally needed a cloud cover index.

Another approach is a distributed photovoltaic systems capacity estimator [4]. The algorithm determines whether a customer has a PV, by using Support Vector Machines (SVM). Four weather status-driven features are extracted for SVM, they describe the difference of net charge profiles between customers with and without PV - the ratio of total electricity consumption, concave shape index, concavity degree and load ramping rate. This method needs the PV power output of the known customers, which is its main drawback.
Finally, authors in [5] present a method for detecting and disaggregating behind-the-meter solar generation using weather data, advanced metering infrastructure, substation monitoring and generation monitoring from selected nearby PV systems. Similarly, the needs of that method concern power from other nearby PV.
The above methods rely on various types of data such as satellite imagery, solar radiation data, data from other PV stations or detailed weather data to be operational. The method proposed in this paper, on the contrary, does not require such data and does not contain any detailed modelling. It strictly relies on smart metering and temperature data. In doing so, the proposed approach avoids user privacy issues that have not been addressed previously and only uses aggregated consumption data at the feeder level, as presented below.

Machine learning techniques as neural networks (NN) could provide a large number of functions in the energy field [6], the most popular being forecasting and disaggregation. To the best of our knowledge, a PV generation detection method that combines the trade-offs between data limitation, simplicity of implementation and usable results has not been extensively covered. In this paper, the approach of time series forecasting based on NN is applied to a classification task. The proposed method combines a conventional Multi-Layer Perceptron (MLP) together with an analytical classification algorithm.
The paper is organized as follows: Section II describes the data and simulation environment for the experiment and presents the method. The obtained results and sensitivity analysis of this method are given in Section III. Finaly, conclusions are drawn in Section IV.

2. Methodology

2.1 Experimental setup

The Pandapower tool was used to simulate the electrical distribution grid [7]. It is a Python-based tool combining the “pandas” data analysis library and the “pypower” power-flow solver to create a grid calculation program, aimed at automating analysis and optimization in electrical energy systems. The simulated grid used in the study is the CIGRE-Network medium voltage distribution with 14 nodes (Fig. 1).
The publicly available consumption data "Smart meters in London” were used as hourly net consumption profiles of the simulated grid for two consecutive years (called year *n-1* and year *n*) [8]. This database contains energy consumption for 5567 London households which took part in the “Low Carbon London” project led by British grid operators. The data covers period between November 2011 and February 2014. With a large number of real consumers over a large period of time, (and with virtually no lost measurement points), this dataset is perfectly suited for experiments. Data for the complete 2012 and 2013 years were extracted. Almost 900 consumption profiles were aggregated into 14 groups for each node of simulated grid (around fifty households per node).
The data of solar radiation (typical meteorological year, TMY) for London from web application NREL's PVWatts [9] was used to model PV generation in some nodes. To get the temperature, the data DarkSky API [10] was used for the same geographic position and datetimes as the net consumption. PV generation profiles, modeled using solar radiation data, were added to seven random nodes in year *n* and scaled according to the objectives of the experiments. However, the solar radiation is used only for modeling and not as exogenous data for the detection.
In this paper, the value of the PV installed capacity in the node is expressed as a ratio to the peak load value of this node $P_{nom}^{PV} / \max(P^{load})$. There are 14 nodes for the simulation over two years, seven of which (arbitrary chosen) have PV added during the second year (Fig. 1). The goal is to develop an approach that can correctly detect the PV connection to these seven nodes of a distribution grid during a considered period (e.g. during the last year, month, or any other duration).

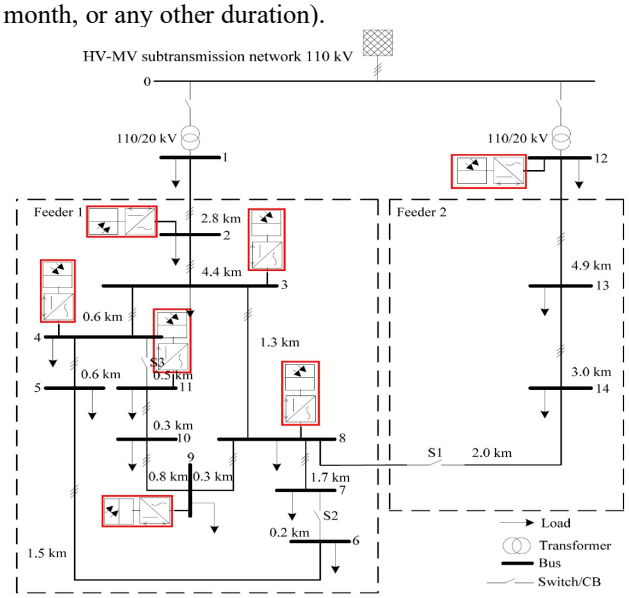

Fig. 1. Medium voltage distribution network with PV (in red) installed in seven random nodes [7].

2.2. Method

The developed tool consists of a NN coupled with an analytical classification algorithm that operates separately for each node. Thus, it is not necessary to take into account the grid topology. However, Pandapower needs a topology

to run a simulation and get the consumption-n profiles for each node. The operating principle of the method is shown in Fig. 2. The data in the first red rectangle (at the bottom) represents the year *n-1* ("training set") and the data in the second (at the top) represents the year *n*, for which the recently installed PV shall be detected.

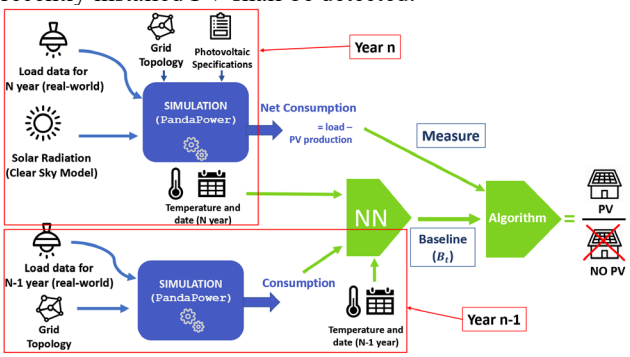

Fig. 2. Principle of operation of the method.

Grid topology and real consumption data for *n-1* year are first integrated into the Pandapower simulator. The simulator then generates hourly energy consumption per node. Besides, the corresponding temperature, time, and date data are added for each point of power consumption. Trained with those data for the year *n-1*, the NN produces the expected consumption for a given day and temperature of the year *n* at an hourly resolution, assuming that no new PV has been added. This forecast is called *baseline,* which is then fed to the classification algorithm.

The net energy consumption measured by the meters (which measure the difference between load and PV generation) for the considered period of the year *n* is also supplied to the classification algorithm and is called *measurement*. Similar to the data for the NN training, those measurements were generated with the Pandapower tool, by taking into account the consumption of households and solar radiation.

Finally, the analytical classification method compares the *baseline* and *measurement* at the same hours throughout the day and then during only the sunshine in order to detect PV. After sensitivity studies, five features were selected for NN that show the strongest effect on consumption. The selected features were converted by hot encoding into 31 neurons of the first NN layer (Fig. 3).

One hot encoding is a process by which categorical variables are converted into a form of bits, i.e., if a data point belongs to the category then corresponding component is 1, otherwise it is 0. In considered NN:

- 24 neurons indicate the hour of the day for what the simulation will be performed, h_i, where $i = 1. \ldots 24$;
- Four others indicate the season, S_h (winter), S_p (spring), S_e (summer), S_a (autumn);
- The last three are temperature ($T°$), weekend (W) and holiday (F).

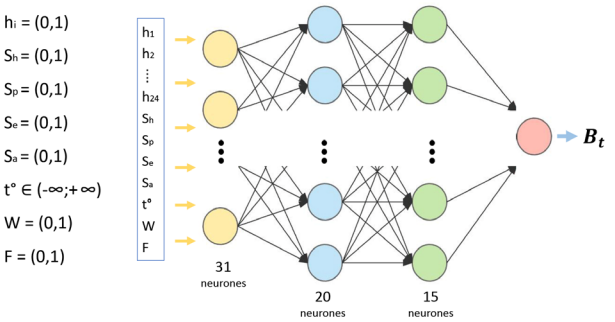

Fig. 3. The neural network and its features.

The NN architecture with two hidden layers, of 20 and 15 neurons each, has shown the highest accuracy as the result of sensitivity studies. The output layer consists of only one neuron, which gives the expected consumption of the node for the chosen hour, given that no PV is connected. The NN is trained with the following parameters: Adam optimizer, sigmoid activation function for the all nodes, mean absolute error loss function, learning rate of 0.03 and 1000 epochs. The training time is 10-20 seconds per node.

The analytical classification algorithm is presented in Fig. 4. The basic idea is that, in absence of PV generation, an average difference between the *baseline* and the *measurement* values is roughly the same for hours of the day and of the night. Thereby, if this difference is greater during hours of the day, the algorithm detects the installation of new PV in the considered period, since the PV generation decreases the *measurement* values only during sunshine.

First, the algorithm calculates E_t (in %) as the difference between the *baseline* (B_t) and the *measurement* (P_t), for each hour t of period T.

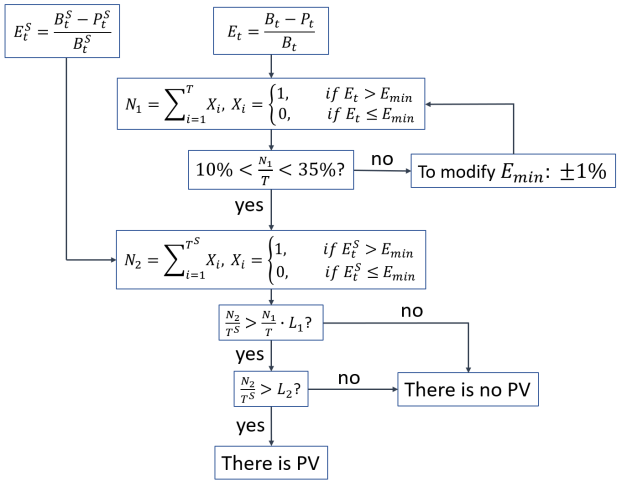

Fig. 4. Synoptic of the detection algorithm.

Then E_t^S is similarly computed as the difference between the *baseline* and the *measurement*, but only for sunshine hours (between 9 a.m. and 4 p.m. of each day). Thereby the total considered period T^S is equal to 2920 hours for the whole year or less if $T < 8760$ hours. The time period 9 am - 4 pm is selected as period with noticeable solar radiation level whatever the season.

The algorithm then counts the number of hours N_1 when E_t exceeds the threshold value E_{min} (10% by default). In other words, it counts the number of hours for which the *baseline* is at least 10% greater than the *measurement* result.

If N_1 is outside the interval of [10 %; 35 %] of the total number of hours in the considered period, the algorithm adjusts the threshold E_{min} by 1 % to avoid an overestimation or underestimation from the NN and then repeats the process, counting N_1 again until N_1 falls into the interval. E_{min} is the only threshold that needs adapting. The values of the interval were obtained empirically through sensitivity studies that experimentally determined the best percentages.

Once E_{min} is set, the algorithm calculates N_2, the number of hours for which the *baseline* is greater than the *measurement* during the period T^S by at least the value of the previously obtained threshold E_{min}.

Then the algorithm checks if N_2/T^S is at least $L_1(140\%)$ times greater than N_1/T. Thus, it checks that the hours when the *baseline* is greater than the *measurement* on E_{min} are more frequent during the sunshine (period T^S) than during the whole period T. If there is no new installed PV, N_2/T^S and N_1/T should be roughly the same. If there is new PV, then N_2/T^S will be significantly greater than N_1/T.

Finally, the algorithm checks that N_2/T^S is greater than L_2. That is, at least during L_2 (35 %) hours of T^S *baseline* is greater than *measurement* by E_{min}%. This is necessary to avoid situations where, for instance, $N_1/T = 10$ % and $N_2/T^S = 15$ %. Indeed, then, the previous condition is met, but there is no PV on the node, because the proportion of N_2/T^S is too small. If both of the last checks were successful, the algorithm concludes that there is a new PV installed in this node.

3 Results

As already mentioned, 14 nodes with different consumption profiles were prepared, with PV connected to seven of them, arbitrary selected (Fig. 1). The proposed tool was independently tested on each of these 14 nodes.

It is possible to increase the sensibility of the tool by analyzing only months with high level of solar radiation instead of the whole year. To confirm this hypothesis, two different cases are considered: application of the tool with consumption data over the six sunniest months (from April to September) and over the six least sunny months (from October to March).

The average accuracy was calculated for all possible periods of x days over 6 months with a rolling basis. (e.g. for "period = 40 days", the average accuracy of 142 possible 40-day periods between April and September was calculated).

The dependence of the average accuracy of the tool with respect to the period under consideration and to $P_{nom}^{PV} / \max(P^{load})$ for all nodes of the grid is presented in Fig. 5 (from April to September) and in Fig. 6 (from October to March).

The average accuracy from April to September is between 0.8 ($P_{nom}^{PV} / \max(P^{load}) = 4.5$ %, over a period of five days) and 1 ($P_{nom}^{PV} / \max(P^{load}) > 7.0$ %, for a period of more than two months).

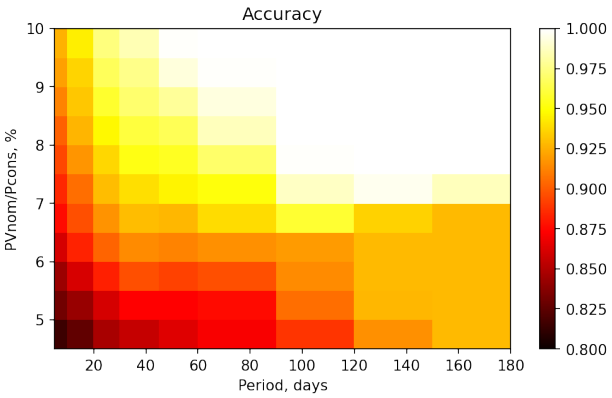

Fig. 5. Average accuracy dependence from April to September.

It can be concluded, that the average accuracy for the sunniest months is higher for longer periods, because the longer period allows to smooth out the impact of cloudy days, when PV generates less energy. It is also natural that a higher $P_{nom}^{PV} / \max(P^{load})$ makes it easier to detect PV, so the average accuracy is also higher.

The results are different from October to Mars. The average accuracy for these months is between 0.87 ($P_{nom}^{PV} / \max(P^{load}) = 10.0$ %, over a period of five months) and 0.5 ($P_{nom}^{PV} / \max(P^{load}) = 4.5$ %, for a period of six months). For values $P_{nom}^{PV} / \max(P^{load}) < 8.5$ % the average accuracy is higher for shorter periods, because on average during these months the PV systems do not generate enough power for detection, but there are few days with high solar radiation level when detection is possible.

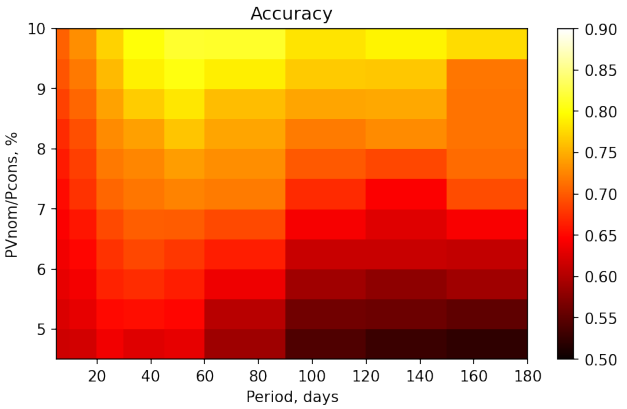

Fig. 6. Average accuracy dependence from October to March.

From Fig. 5 and Fig. 6 it can be concluded that, on average, three months are required to obtain the maximum detection accuracy. Considering the periods of the three sunniest months, 100 % accuracy can be achieved with values of

6.2 % for the installed PV capacity in the node ($P_{nom}^{PV} / \max(P^{load})$) and 3.1 % for the generated PV energy compared to the energy consumption of the same node (W_{nom}^{PV} / W^{load}), as shown in Table 1.

Table 1. Distribution of the installed PV power and the corresponding energy production.

Bus	2	3	4	8	9	11	12
$\frac{P_{nom}^{PV}}{\max(P^{load})}$,%	6,2	6,2	6,2	6,2	6,2	6,2	6,2
$\frac{W_{nom}^{PV}}{W^{load}}$,%	3,25	3,2	3,3	3,1	3,05	3,05	3,68

The results of the metrics (for nodes with PV) that are used to detect PV are presented in Table 2. Data are given for the three sunniest months of the year (from May to July). The value of $\frac{N_1}{T} / \frac{N_2}{T^S}$ for node 12 is equal 1.4, which means that further decreasing the installed PV capacity in that node will cause the tool to be unable to detect it.

Table 2. Results for nodes with PV from May to July 2013.

Bus	2	3	4	8	9	11	12
$\frac{N_1}{T}$	34,4	33,6	34,6	33,5	34	33,5	32,8
$\frac{N_2}{T^S}$	50,6	59,3	60,1	57	62,3	54	46
$\frac{N_1}{T} / \frac{N_2}{T^S}$	1,47	1,76	1,74	1,70	1,83	1,61	**1,40**

It should be mentioned that the sensitivity of the tool depends on consumption profiles (for two years), so for Node 11, for example, the tool can detect PV even with very small installed capacity ($W_{nom}^{PV} / W^{load} < 3$ %).

4 Conclusion

The growing need for accurate prediction of consumption in distribution grid requires a way to detect locations with "hidden'' installed PV. The use of neural networks for capturing energy consumption behavior is very popular as the model can be trained offline and then quickly applied for any period of time.

The method proposed in the article for PV detection only needs data about temperature and net consumption that makes it more suitable for usage than other similar solutions, which require specific hard-to-find data (e.g., satellite images or solar radiation for specific spot). Moreover, the algorithm displays a high accuracy, which however depends on the period of consideration and the amount of installed capacity. Thus, it is necessary to find a compromise between the volume of available data and the required accuracy of the solution. It was found that the highest precision and accuracy could be reached for approximately a three months period under consideration. Future work may involve testing the tool on larger simulated grids. The next step will then be to not only detect PV installations in the nodes, but also to disaggregate the load and generation profiles, and hence approximate the installed PV capacity. Regarding the data, additional consideration could be given to noise and anomalies, because the simulated data has been very clean in the presented work.

5 References

[1] Donaldson D. L, Jayaweera D.: 'Effective solar prosumer identification using net smart meter data'. Int J Electric Power Energy Syst 2020;118:105823. https://doi.org/10.1016/j.ijepes.2020.105823. ISSN 0142-0615.

[2] Malof JM, Bradbury K, Collins LM et al.: 'Automatic detection of solar photovoltaic arrays in high resolution aerial imagery'. Appl Energy 2016;183:229–40.

[3] Zhang X, Grijalva S.: 'A data-driven approach for detection and estimation of residential PV installations'. IEEE Trans Smart Grid;7(5):2477–85, 2016

[4] Wang F, Li K, Wang X et al.: 'A distributed PV system capacity estimation approach based on support vector machine with customer net load curve features'. Energies; 11(7):1750, 2018.

[5] Michaelangelo T., Sila K. and Emre C. K.: 'Disaggregating solar generation behind individual meters in real time'. In Proceedings of the 5th Conference on Systems for Built Environments. ACM, 43–52, 2018.

[6] Ali S, Choi B.J.: 'State-of-the-Art Artificial Intelligence Techniques for Distributed Smart Grids: A Review'. Electronics, 9, 1030, 2020.

[7] 'CIGRE Networks', https://pandapower.readthedocs.io/en/v2.5.0/networks/cigre.html, accessed 13 December 2020.

[8] 'Smart meters in London', https://www.kaggle.com/jeanmidev/smart-meters-in-london, accessed 13 December 2020.

[9] 'PVWatts Calculator', https://pvwatts.nrel.gov/index.php, accessed 13 December 2020.

[10]'Dark Sky API', https://darksky.net/dev, accessed 13 December 2020.