A TCN-Based Hybrid Forecasting Framework for Hours-Ahead Utility-Scale PV Forecasting

Yiyan Li, Member, IEEE, Lidong Song, Student Member, IEEE, Si Zhang, Student Member, IEEE, Laura Kraus, Taylor Adcox, Roger Willardson, Abhishek Komandur, and Ning Lu, Fellow, IEEE

Abstract—This paper presents a Temporal Convolutional Network (TCN) based hybrid PV forecasting framework for enhancing hours-ahead utility-scale PV forecasting. The hybrid framework consists of two forecasting models: a physics-based trend forecasting (TF) model and a data-driven fluctuation forecasting (FF) model. Three TCNs are integrated in the framework for: i) blending the inputs from different Numerical Weather Prediction sources for the TF model to achieve superior performance on forecasting hourly PV profiles, ii) capturing spatial-temporal correlations between detector sites and the target site in the FF model to achieve more accurate forecast of intra-hour PV power drops, and iii) reconciling TF and FF results to obtain coherent hours-ahead PV forecast with both hourly trends and intra-hour fluctuations well preserved. To automatically identify the most contributive neighboring sites for forming a detector network, a scenario-based correlation analysis method is developed, which significantly improves the capability of the FF model on capturing large power fluctuations caused by cloud movements. The framework is developed, tested, and validated using actual PV data collected from 95 PV farms in North Carolina. Simulation results show that the performance of 6 hours ahead PV power forecasting is improved by 20% - 30% compared with state-of-the-art methods.

Index Terms—Detector network, neighbor selection, NWP blending, physics-based model, short-term PV forecast, spatial-temporal forecasting, temporal convolutional network.

NOMENCLATURE

Scalar

| Symbol | Definition |
|--------|------------|
| \(d\) | Dilation rate |
| \(D\) | Number of days in the historical data |

Functions

| Symbol | Definition |
|--------|------------|
| \(E_{bias}\) | Bias in the physics-based model |
| \(h\) | Index of the historical days |
| \(K\) | Filter size |
| \(M\) | Length of the daily irradiance profile |
| \(N\) | Number of PV sites |
| \(N_{stack}\) | Number of module stacks in the TCN model |
| \(P_x\) | Actual power output from field measurements |
| \(P_{simu}\) | Power output of the physics-based model at time \(t\) |
| \(P_{cc}\) | Pearson Correlation Coefficient |
| \(P_{cc,\text{max}}\) | \(P_{cc}\) value with the optimal time shift, \(\Delta t_{\text{max}}\) |
| \(P_{x,y}\) | Receptive field of the TCN model |
| \(S_h\) | Index of the correlation scenarios on \(h\)th day |
| \(t\) | Index of the time series data |
| \(\Delta t\) | Time shift between two time series |
| \(\Delta t_{\text{max}}\) | Optimal time shift that leads to \(P_{cc,\text{max}}\) |
| \(T_{shift}\) | Threshold of time-lagged correlation analysis |
| \(T_{thre}\) | Threshold of \(\Delta t_{\text{max}}\) to determine successful detection |
| \(\Delta x\) | Irradiance change threshold to detect cloud event |
| \(\varphi\) | Successful detection rate |
| \(\varphi_{\text{max}}\) | Maximum successful detection rate |

Vector/Matrix

| Symbol | Definition |
|--------|------------|
| \(F\) | Filter in TCN, \(F = [f_0, f_1, \ldots, f_{k-1}]\) |
| \(X\) | Normalized irradiance vector, \(X = [x_1, x_2, \ldots, x_M]\) |
| \(X_T\) | \(X\) of the target site, \(X_T = [x_{T1}, x_{T2}, \ldots, x_{TM}]\) |
| \(X_D\) | \(X\) of the detector site, \(X_D = [x_{D1}, x_{D2}, \ldots, x_{DM}]\) |
| \(\hat{X}_T\) | Differential vector of \(X_T\), \(\hat{X}_T = [\Delta x_T1, \Delta x_T2, \ldots, \Delta x_TM]\) |
| \(\hat{X}_D\) | Differential vector of \(X_D\), \(\hat{X}_D = [\Delta x_D1, \Delta x_D2, \ldots, \Delta x_DM]\) |
| \(F\) | Vector of the neighboring sites |
| \(F_{\text{opt}}\) | Vector of the selected optimal detector network |
| \(T\) | Matrix of \(\Delta t_{\text{max}}\), \(D \times (N-1)\) |
| \(P\) | Matrix of \(P_{cc,\text{max}}\) |
| \(\Phi\) | Vector of \(\varphi\), \(1 \times (N-1)\) |

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSG.2023.3236992.
Digital Object Identifier 10.1109/TSG.2023.3236992

1949-3053 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
I. INTRODUCTION

The stochasticity of cloud movements is the major cause for large PV forecasting errors, especially in the hours-ahead time horizon. Existing PV forecasting methods can be divided into physics-based models and data-driven models [1], [2], [3]. Physics-based models use irradiance forecasts (usually obtained from Numerical Weather Prediction (NWP), satellite images, total sky imagers, etc.) as inputs to forecast PV power outputs. Once a physics-based model is well-calibrated, its forecasting accuracy and resolution will completely rely on the forecasting accuracy of its input data. Meanwhile, because the physics-based model requires detailed parameters of PV modules/inverters, it requires considerable efforts for maintaining model accuracy. Therefore, the physics-based modeling approach is usually used for modeling MW-level PV farms and rarely applied to residential rooftop PV systems.

Data-driven models do not require parameters from PV system components. Methods, such as regression or machine learning based models, can be used for predicting the future PV power output from historical data assuming that the future follows similar patterns as the past. Because weather conditions can change rapidly in a day, the data-driven model is usually used for hours-ahead forecasting while the physics-based methods for day-ahead PV forecasting. The two methods are further compared in Table I.

Recently, leveraging spatial-temporal correlations between neighboring sites for capturing cloud movements to improve short-term PV forecasting accuracy, as an emerging branch of data-driven method, has drawn increasing attention. By analyzing power outputs of a group of PV sites or weather data from adjacent meteorological stations, patterns of cloud movements can be accounted for implicitly or explicitly. Regression models, such as AR, ARX, NARX, are firstly introduced in [4], [5], [6], [7] to build the mapping between neighboring sites and the target site. In [8], Yang et al. achieve spatial-temporal PV forecasting using time-forward kriging. In [9], Kim and Lee use spatial-temporal Kriging to estimate the irradiance at the target location based on the measurement data from nearby weather stations, and then build a probabilistic model to forecast the solar power at the target PV farm. Compressive sensing is used by Tascikaraoglu et al. in [10] to extract the spatial-temporal correlations between a target meteorological station and its neighboring stations to improve the short-term forecasting accuracy. In [11], Zhang et al. use Bayesian network to forecast PV generation based on spatial-temporal analysis results. Recently, deep learning receives increasing attention because of its powerful nonlinear learning ability. In [12], Liu et al. combine CNN and GRU to extract the spatial-temporal information from multi-dimensional time series inputs from adjacent sites. Similarly, in [13], Wang et al. implement CNN to extract the spatial features across multiple sites, and then use LSTM to achieve the forecasting of the target site. In [14], Jeong and Kim organize the multi-site time series data into spatial-temporal matrix, and then use CNN to achieve feature extraction and forecasting. In [15], Venugopal et al. compare different CNN structures in extracting features from heterogeneous data sources, as well as their short-term PV forecasting performances. Compared with the forecasting methods using total sky imagers or satellite images, spatial-temporal forecasting methods are usually cheaper and easier to implement while achieving comparable performance without requiring extensive hardware or external data supports.

However, we identify two research gaps in the state-of-the-art methods. First, there lacks of a deep fusion approach for integrating physics-based and data-driven models into a hybrid PV forecasting framework to leverage both of their advantages. As shown in Table I, the physics-based model can achieve stable hourly irradiance-power conversion for a much longer forecasting horizon than the data-driven model. However, it cannot effectively capture intra-hour power fluctuations caused by cloud movements. The data-driven model can forecast intra-hour PV fluctuations with higher accuracy by capturing cloud movements from near real-time measurements. However, the forecasting accuracy and stability of data-driven model decrease rapidly when the forecasting horizon is longer than a few hours. Thus, the deep fusion between the physics-based model and the data-driven model has the potential to achieve both forecasting stability and finer granularity. To the best of the author’s knowledge, such studies are rarely seen in existing literatures.

Second, there lacks an automated, objective detector site selection mechanism. Selecting the most contributive neighbors, i.e., the “detector” sites, for the “target” site among all its neighbors is crucial for improving forecasting accuracy and reducing model complexity in spatial-temporal forecasting models. Conventional detector site selection methods ignore the sequence of cloud events (i.e., which site sees the cloud event first and which next). In addition, existing methods are pairwise-correlation-based and cannot exploit the benefit of “collaboration”, where a certain combination of neighboring sites can formulate a more powerful detector network. Lastly, conventional methods usually involve using manually-defined correlation thresholds for eliminating “uncorrelated” neighbors, making the process highly subjective.

Therefore, in this paper, we develop a Temporal Convolutional Network (TCN) based hybrid PV forecasting framework for enhancing hours-ahead utility-scale PV forecasting. The hybrid framework consists of two forecasting models: a physics-based trend forecasting (TF) model and a data-driven fluctuation forecasting (FF) model. Three TCNs are integrated in the framework: in the TF model, the first TCN is used for blending TF input data from different NWP
Fig. 1. Flowchart of the proposed hybrid PV forecasting framework.

sources into a coherent input data set that are then fed into the inverter-level physics-based model to obtain more accurate trend forecasting results. In the FF model, the second TCN is used for hours-ahead, intra-hour PV power forecasting, where predicting power fluctuations caused by large cloud events is crucial. Using measurements from selected detector sites as inputs, the second TCN can generate more accurate intra-hour PV forecasts with minute-level granularity. Then, the third TCN is used to reconcile the TF and FF results. Using a sequence-to-sequence model for achieving temporal consistency between the two time series forecasts, we can further improve the short-term PV forecasting accuracy by capturing both the hourly trends and intra-hour PV fluctuations.

In summary, the contributions of this paper are two-folds. First, the proposed hybrid PV forecasting framework combines the advantages of the physics-based model and the data-driven model so that both the hourly trend and the intra-hour fluctuations can be well predicted. Second, we develop a scenario-based, automated detector site selection algorithm to identify the most contributive neighbors for a target site. One distinct advantage of the detector selection algorithm is that we consider the temporal leading/lagging patterns and the collaborative effect between sites. The proposed algorithm is automated and objective so neither domain expertise nor human supervision is required.

The rest of the paper is organized as follows: Section I introduces the proposed hybrid PV forecasting framework. Section II demonstrates the simulation results. Section III concludes this paper.

II. METHODOLOGY

The proposed hybrid PV forecasting framework for hours-ahead PV forecasting is illustrated in Fig. 1. The framework consists of two forecasting models: TF and FF. The TF model is a physics-based model for predicting hourly PV power outputs over a time horizon of up to a week. The FF model is a data-driven model for predicting intra-hour PV output fluctuations at 5-minute resolution for the next 1 to 6 hours.

The input of the TF model is the hourly NWP data. To improve the TF model performance, a TCN-based NWP data blender (TCN #1, detailed in Section II-B) is developed to fuse the data from different NWP sources with an objective of maximizing the forecast accuracy of irradiance inputs. Then, the predicted irradiance from TCN is converted into hourly PV forecasts by the physics-based model.

The inputs of the FF model are the normalized historical irradiance data from the PV sites in the detector network. To select the most-correlated neighbors to form the detector network, we develop a scenario-based neighbor selection algorithm for automatically identifying an effective detector network for the target site without human supervision (detailed in Section II-C). First, at each time step, the irradiance data collected at all PV sites will be coded into a two-dimensional (2D) matrix based on their geographical locations. Then, the historical data of the target site together with the detector sites will be fed into the TCN-based FF model (TCN #2) to extract the spatial-temporal information to produce the intra-hour PV forecasting.

To perform forecast reconciliation, forecasting results from both TF and FF models will be fed to a TCN (TCN #3, detailed in Section II-D) to eliminate the inconsistencies (i.e., different data resolution and magnitude discrepancies) between the two time-series profiles. The resultant hours-ahead PV forecast with 5-minute resolution captures both the hourly trend and intra-hour fluctuations by combining the advantages of
two independent forecasting models, namely the physics-based model and the data-driven model.

In practice, the TF model can be updated every 6 hours (same as the NWP data update cycle) while the FF model can be updated every 5-minute or longer depending on the communication network setup between the control center and each PV site.

A. Temporal Convolutional Network

TCN is a fully convolutional-based network structure [16], as shown in Fig. 2. Each convolutional layer needs to have the same length as the input layer. To meet this requirement, zero padding is applied to solve the dimension reduction issue caused by the convolution operation (see the dashed blocks in Fig. 2). After a few convolutional layers, the features of the input time series are extracted and compressed into the output layer, which can be further used for forecasting purpose. Compared with sequential networks such as RNN and LSTM, such a purely convolutional structure of TCN can be highly paralleled in model training and therefore has better training efficiency [16]. TCN has been successfully used to solve the time series forecasting problems, such as PM2.5 forecasting [17] and load forecasting [18], etc. However, it has not yet been introduced for conducting PV forecasting.

The most significant feature of TCN is the dilated convolution. Assume the input time series is \( X = [x^0, x^1, \ldots, x^M] \), and we use a filter \( F = [f_0, f_1, \ldots, f_{K-1}] \) to conduct convolution. Then, the dilated convolution \( G(\cdot) \) for the element \( x^m \) in \( X \) can be calculated by

\[
G(x^m) = \sum_{i=0}^{K-1} f_i \cdot x^{m-\lfloor d \rfloor i}
\]  

(1)

where \( d \) is the dilation rate, and \( m-\lfloor d \rfloor i \) indexes to the past historical data before \( x^m \). When \( d = 1 \), (1) reduces to a regular convolution operation (e.g., the first layer in Fig. 2). When \( d \) is larger than 1 (e.g., the second and third layers in Fig. 2), the filter will skip \((d-1)/d\) of the elements in the previous hidden layer and only focus on the remaining \(1/d\). In this way, the whole network will have a large receptive field that increases quickly with the number of layers with limited model complexity. The receptive field of TCN model can be calculated by

\[
R_{field} = 1 + 2 \cdot (K - 1) \cdot N_{stack} \cdot \sum_i d_i
\]  

(2)

According to (2), we have 3 ways to increase the receptive field: using larger filter size \( K \), larger dilation rate \( d \), and increasing the network depth \( N_{stack} \).

Causal convolution is another feature of the TCN model. As shown in Fig. 2, each convolution layer only extracts information from the past historical data. In other words, there is no “information leakage” from the future. This structure is particularly suitable in solving forecasting problems where the future information is unavailable. To further improve the model performance, residual connections [19] can be used to achieve identical mappings.

B. TCN-Based NWP Data Blending

In practice, inverter-level physics-based models are commonly used to convert the irradiance (obtained from the NWP data) to power based on the inverter used at each PV site. However, the NWP data can come from different sources, for example, High-Resolution Rapid Refresh (HRRR), Global Forecast System (GFS), National Digital Forecast Database (NDFD), Rapid Refresh (RAP) and North American Mesoscale (NAM). As shown in Table II, because different NWP data sources have different forecasting features and data granularity, when using different NWP data as inputs, discrepancy in TF model arises.

This inspires us to develop a data blending model to merge the NWP data from different sources together with an objective to offset modeling deficiencies and improve the overall forecasting performance. Thus, we developed a TCN-base NWP data blending model (See TCN #1 in Fig. 1) to achieve the seq2seq mapping from NWP to the actual irradiance. The inputs are time-series data from different NWP data sources, and the output is the time series of the field measurement irradiance.

C. Scenario-Based Detector Sites Selection Algorithm

The main challenge in short-term PV forecast is to forecast, in a few hours-ahead, the large power drops caused by cloud movements. Thus, the objective of the detector sites selection algorithm is to select an optimal group of detector sites so that using the solar irradiance data of those detector sites as inputs, large PV output drops at the target site can be more accurately predicted. Note that, in this paper, we focus our analysis on
detecting cloud events instead of the entire irradiance time series. As shown in Fig. 3(a), the irradiance data used in our study is in 5-minute granularity. Thus, there are 288 data points in a day (i.e., \( M = 288 \)). Normalize all the daily irradiance profiles for all PV sites. Let the target site irradiance profile be \( X_T = [x_{T1}, x_{T2}, \ldots, x_{T M}] \) and the detector site be \( X_D = [x_{D1}, x_{D2}, \ldots, x_{D M}] \).

Define a cloud event as the irradiance drop greater than threshold \( \Delta x \) during two consecutive time intervals

\[
x_t - x_{t-1} \leq -\Delta x
\]

Note that \( \Delta x \) is dependent on data granularity. In this paper, we define \( \Delta x = 0.3 \). This is because we want to capture the cloud event that can cause a power drop greater than 30% of the rated power in 5 minutes.

To extract the cloud event for the target site and the detector site, \( \hat{X}_T \) and \( \hat{X}_D \), respectively, we have

\[
\begin{align*}
\hat{X}_T & = \begin{bmatrix} \hat{x}_{T1}, \hat{x}_{T2}, \ldots, \hat{x}_{TM-1} \end{bmatrix} \\
\hat{X}_D & = \begin{bmatrix} \hat{x}_{D1}, \hat{x}_{D2}, \ldots, \hat{x}_{DM-1} \end{bmatrix} \\
\hat{x}_{Ti}, \hat{x}_{Di} & = \begin{cases} 0, & \hat{x}_{Ti}, \hat{x}_{Di} > -\Delta x \\ \hat{x}_{Ti}, \hat{x}_{Di} \leq -\Delta x & \end{cases}
\end{align*}
\]

From the irradiance profiles of the two PV sites shown in Fig. 3(a), we can extract the cloud event series of the detector and target sites as shown by the solid lines in Fig. 3(b) and 3(c).

Next, a time-lagged correlation analysis [20] is conducted to find the optimal time shift \( \Delta t_{\text{max}} \) between the target and the detector series so that the Pearson Correlation Coefficient [21] between the two sites, \( P_{cc} \), can be maximized. The problem is formulated as

\[
\Delta t_{\text{max}} = \text{argmax}_{-T_{\text{shift}} \leq \Delta t \leq T_{\text{shift}}} P_{cc}(\hat{X}_T[\Delta t : M + \Delta t - 1], \hat{X}_D)  
\]

The indexed values out of range \([1 : M - 1]\) are padded by 0. In (6), we select \( T_{\text{shift}} = 8 \)h. Note that \( T_{\text{shift}} \) is a constant threshold of \( \Delta t \) that guarantees the correlation calculation is within the same-day horizon. As shown in Fig. 3(c), after \( \hat{X}_T \) is shifted by \( \Delta t_{\text{max}} \) (the dotted line), it is well aligned with \( \hat{X}_D \).

After calculating the \( P_{cc} \) between the target site and detector sites, existing methods [5], [11] select detectors with the highest \( P_{cc} \) values under the assumption that a site with high \( P_{cc} \) indicates higher contribution to the forecasting accuracy of the target site. However, such methods have three drawbacks:

- The approach ignores the temporal correlation crucial to detect cloud events. Only sites with a leading correlation with the target site will contribute to the target site forecasting results. This is because only when the cloud passes the detector site earlier than the target site (\( \Delta t_{\text{max}} > 0 \)), can the information be used to forecast the target site cloud events. A detector site having a high \( P_{cc} \) but with lagging correlation (\( \Delta t_{\text{max}} \leq 0 \)) cannot foresee the upcoming cloud event on the target site.
- The selection of the \( P_{cc} \) threshold is subjective and empirical. In practice, different neighbor sites have different \( P_{cc} \) with the target site. A \( P_{cc} \) threshold is therefore needed to eliminate low-correlated neighbors. However, because the \( P_{cc} \) value can vary significantly for different cases, it is difficult to manually select optimal thresholds or optimal number of detector sites.
- When detector sites are selected pair-by-pair using \( P_{cc} \), the synergetic effect among detectors in a network setting cannot be properly accounted for.

Therefore, in this paper, we propose a scenario-based detector sites selection algorithm to overcome those drawbacks.

First, all the 6 possible correlation scenarios between the target and the neighbor sites are summarized in Table III, based on their daily cloud conditions and \( \Delta t_{\text{max}} \). If there is at least one cloud event during the day, this day is defined as cloudy. Otherwise, this day is defined as clear sky. Note that Scenario 1 is not considered because our focus is to detect the cloud events that will significantly impact the PV output.

| Scenario No. | Detector site | Target site | \( \Delta t \) | Definition |
|--------------|---------------|-------------|----------------|------------|
| 1            | Clear sky     | Clear sky   | \( \| \)         | Ignored    |
| 2            | Cloudy        | Clear sky   | \( \| \)         | Wrongly detect |
| 3            | Clear sky     | Cloudy      | \( \| \)         | Fails to detect |
| 4            | Cloudy        | Cloudy      | \( \Delta t_{\text{max}} \geq 0 \) | Fails to detect |
| 5            | Cloudy        | Cloudy      | \( 0 < \Delta t_{\text{max}} \leq T_{\text{thre}} \) | Successful detection |
| 6            | Cloudy        | Cloudy      | \( T_{\text{thre}} < \Delta t_{\text{max}} \) | Irrelevant |

Among the six scenarios, Scenario 5 is defined as “successful detection”. This means that a cloud event occurs earlier in the detector site than in the target site with a leading time \( \Delta t_{\text{max}} \in (0, T_{\text{thre}}] \). Note that \( T_{\text{thre}} \) equals to the forecasting horizon in the FF model to pick out the most contributively neighbors (e.g., in a 2-hour ahead forecasting scenario, \( T_{\text{thre}} = 2 \)).

Based on Table III, we further define the successful detection rate for a detector site as

\[
\varphi = \frac{\sum_{h=1}^{D} I(5)(S_h)}{\sum_{h=1}^{D} I(2,3,4,5,6)(S_h)}
\]
where $\mathcal{I}_A(x)$ is the indicator function that

$$
\mathcal{I}_A(x) = \begin{cases} 
1, & \text{if } x \in A \\
0, & \text{if } x \notin A 
\end{cases}
$$

For a given detector-target site pair, $\varphi \in [0, 1]$ measures the successful detection rate. A larger $\varphi$ means the neighbor site has a statistically significant leading correlation pattern with the target site, indicating that the wind direction has a higher chance to blow from the detector site to the target site. Thus, $\varphi$ is an indicator that reflects localized weather patterns dominated by geographical characteristics.

To maximize $\varphi$, a detector network containing multiple neighbor sites is needed to have a better chance to forecast the upcoming cloud events for the target site. When calculating $\varphi$ of a detector network, for each historical day we select the neighbor with the largest $\Delta t_{\text{max}}$ as a representative because it has the best prediction ability. However, selecting an optimal subset among all candidate neighbors is a typical NP-hard problem. Therefore, in this paper, we design a greedy-searching algorithm to find a near-optimal solution. Key steps of the algorithm are summarized as follows:

**Step 1 (Data Preparation):** Calculate the maximum time-lagged correlation coefficients, $P_{c,\text{max}}$, and the corresponding $\Delta t_{\text{max}}$ between each candidate neighbor site and the target site based on historical cloud event data.

**Step 2 (Forming the Detector Network):** Add each neighboring site successively to the detector network in descending order base on yearly averaged $P_{c,\text{max}}$. Every time a new neighbor is included, recalculate $\varphi$ for the detector network. After all potential neighbors are added into the detector network, the detector network with the maximum $\varphi$ value will be selected.

**Step 3 (Network Refinement):** The goal of refinement is to remove “bad” neighbors by successively removing the neighbors in the detector network to see if $\varphi$ can be further improved. If $\varphi$ is improved after a “bad” neighbor is removed, go back to step 2) and redo the detector network forming with the bad sites removed. If $\varphi$ cannot be improved, algorithm stops.

The pseudocode of the algorithm is shown in Algorithm 1. After selecting the detector network, we will further put the historical data of both the target site and the detector network into the TCN model to extract their spatial-temporal correlations, shown as TCN #2 in Fig. 1. Results will be discussed in Section III-C.

### D. TCN-Based Forecasting Results Reconciliation

After obtaining the forecasting results under both TF and FF models, it is important to reconcile the results for two reasons. First, because different models are used for producing TF and FF forecasts, inconsistency in results are inevitable. For example, the hourly averages of the 5-min FF forecasts may deviate from the TF hourly forecasts. The inconsistency need to be reconciled so that the system operators know which value to use for future hours. Second, on the one hand, using weather data as inputs and taking a physics-based modeling approach, the TF model captures the PV average power output over a longer forecasting horizon well. On the other hand, the data-driven, TCN-based FF model captures intra-hour cloud events well. Thus, reconciliation can achieve mutual-benefit and improve the overall forecasting performance.

Reconciliation was introduced in the 2010s to solve the inconsistency in hierarchical forecasting problems. In the literature, analytical and machine-learning based reconciliation methods were proposed from both spatial and temporal aspects [22], [23], [24], [25], [26], [27]. In this paper, we propose to use the sequence-to-sequence (seq2seq) mode of TCN for reconciliation. As shown in Fig. 4, the TF and FF forecasts are firstly aligned by time so that each hourly TF forecasting point is paired with 12 FF forecasting points (5-min granularity). This serves as the input of the reconciliation model. The
output is compared with the actual 5-min power output profile of the target PV farm during the same time period. The reconciliation model is trained to fine-tune the FF forecasts under the guidance of the TF forecasts. Once the model is trained, the reconciled forecasting results are converted to two consistent sets of 5-minute and hourly PV outputs.

### III. Case Study

In this paper, 5-minute field data collected from 95 utility-scale PV farms in North Carolina from 1/1/2020 to 11/30/2020 are used to develop and verify the performance of the proposed algorithm. Sizes of the PV farms range from 0.4MW to 26.2MW with locations shown in Fig. 5. The available measurements of each site are summarized in Table IV. For each PV site, irradiance measured by the pyranometer is selected to derive cloud movements because pyranometers are placed horizontally, making the measurement more comparable across different PV sites [28]. Missing points in pyranometer measurements (approximately 87% completeness) are patched using the correlations between pyranometer measurements and other irradiance-related measurements (e.g., inverter-level power output). As shown in Fig. 6, all measurements are normalized by their maximum value.

#### A. TF Model: Model Validation

The physics-based models are built based on the module/inverter parameters (examples are shown in Appendix E) provided by the PV farm operator using PVlib toolkit [29]. After the physics-based models are built, we validate their NWP-to-power conversion accuracy by taking the field measurement weather data as the input to see if it can produce the correct power output. The average model bias is

$$E_{bias} = \frac{1}{M} \sum_{i=1}^{M} (p_{simu}^i - p_{real}^i)$$  \hspace{1cm} (9)

where $p_{simu}^i$ is the simulated power output of the physics-based model, $p_{real}^i$ is the ground truth power output, and $M$ is the length of the time series.

4 inverters from different PV farms are randomly selected as an example to validate the model performance. We feed the physics-based models of the 4 inverters with field measurement irradiance, temperature, wind speed in July 2020 [30] to generate $p_{simu}^i$ and compare with $p_{real}^i$. As shown in Fig. 7 and Table V, we can see that the simulation results follow closely to the ground truth with the average bias less than 1%. This means the physics-based models are accurate and can be readily used to produce forecasting results when fed with NWP data.

#### B. TF Model: TCN-Based NWP Data Blending

We obtain the historical NWP data from five different sources (See Table II) from May to October to test the TF model. The historical data are firstly divided into training (70%), validation (10%), and testing (20%). We make sure the testing samples are evenly distributed on different months and are timely aligned with the testing set in the FF model.
TABLE VI

PERFORMANCE COMPARISON OF DIFFERENCE BLENDING METHODS UNDER DIFFERENT WEATHER CONDITIONS

| Blending methods | Sunny | | | | | | | | Cloudy | | | | | | | | | | Rainy | | | |
|------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
|                  | RMSE  | Bias  | RMSE  | Bias  | RMSE  | Bias  |
| None             | 20.50 | 1.51  | 93.80 | 15.03 | 59.98 | 9.44  |
| LR               | 13.77 | 0.57  | 74.03 | -9.88 | 51.32 | -2.10 |
| RF               | 10.94 | 0.33  | 66.90 | 3.20  | 39.71 | 0.75  |
| SVR              | 13.80 | 0.40  | 68.44 | 5.00  | 45.39 | 2.88  |
| MLP              | 12.61 | 0.11  | 71.01 | -3.77 | 40.74 | -1.02 |
| LSTM             | 10.10 | 0.34  | 63.60 | 1.55  | 38.60 | 2.58  |
| TCN              | 10.48 | 0.22  | 57.68 | 0.74  | 32.93 | -0.20 |

Fig. 8. Forecasting results of the proposed TF model under different weather conditions. First row: sunny days, second row: cloudy days, third row: rainy days.

(see Section III-C) as much as possible. The TCN model configuration is shown in Table IX in the Appendix.

We evaluate the TF forecaster in the following three aspects: First, we compare the TF forecaster performance with and without the TCN blending model to show the necessity of the NWP blending process. Second, we compare the TCN blender with five commonly used benchmarking blending methods to show its advantage: Linear Regression (LR), Random Forest (RF), Support Vector Regression (SVR), Multilayer Perceptron (MLP) and LSTM. Third, we evaluate the TF accuracy under different weather conditions: sunny, cloudy and rainy.

Results are shown in the following Table VI. The forecasting results of Leo-B2 inverter under different weather conditions are shown in Fig. 8 as an example to visualize the daily forecasting results. We can see that the TCN blending model has the best performance under cloudy and rainy days and is able to predict the hourly trend of the power outputs.

C. FF Model: TCN-Based Spatial-Temporal Forecasting

In the FF model, we firstly apply the proposed detector site selection algorithm to automatically identify the most contributive neighbors for the target site. The cloud event detection threshold is $\Delta x = 0.3$, and the time-lagging threshold is $T_{\text{thr}} = 1$h (for one hour-ahead forecasting as an example).

As shown in Fig. 9, Leo and Mars are used to demonstrate the proposed scenario-based detector site selection algorithm. At both sites, initially, the successful detection rate $\phi$ will increase when the number of selected neighbors increases. However, when 7 neighbors are selected, $\phi_{\text{max}}$ is reached. This shows that using information from a certain combination of leading-event neighbors helps detecting an upcoming cloud event, while including irrelevant neighbors can pollute the detection accuracy.

After the detector sites are selected, historical data of those sites are used train the TCN model to extract their spatial-temporal correlations for 1-hour ahead forecasting. We compare the TCN model with three state-of-the-art deep learning models that are commonly used for spatial-temporal PV forecasting: Convolutional Neural Network combined with Long-Short Term Memory (CNN-LSTM) [32], VGG-8 (Visual Geometry Group model with 8 layers) [33], [34], and GARNN (Graph Attention Recurrent Neural Network) [35], [36]. To make results comparable, similar model complexity is used, as shown in the Appendix.

The 1-year historical data are split into training (70%), validation (10%), and testing (20%). The simulation is based on i9-9900K CPU with 64GB RAM. We train the models on the training set until they obtain the best performance on the validation set, and then test them on the testing set. All the models are trained with the same settings (i.e., batch size 16, Adam optimizer, Mean Squared Error (MSE) loss, batch normalization) to guarantee fairness.

Violin plots of the 1-hour ahead forecasting RMSE on all the 95 PV sites are shown in Fig. 10 and Table VII. To validate the effectiveness of the neighbor selection algorithm, we train the 4 forecasting models using four kinds of inputs: 1) Selected
Fig. 11. FF forecasting RMSE for the 4 deep-learning models under different forecasting horizons with different detector selection strategies. The bandwidth represents the 90% confident interval (CI-90%) of the forecasting RMSE on 95 PV sites for measuring forecasting stability.

Fig. 12. Comparison of different forecasting models. The black horizontal dashed line shows the forecasting accuracy achieved in the TF by the physics-based model.

TABLE VII
STATISTICS OF THE 1-HOUR AHEAD FORECASTING RMSE ON 95 PV SITES

| Scenarios       | Evaluation Metrics | TCN | CNN-LSTM | VGG-8 | GARNN | Average |
|-----------------|--------------------|-----|----------|-------|-------|---------|
| Selected neighbors | Media              | 27.53 | 29.11    | 29.50 | 27.60 | 28.44   |
| Single site     | Media              | 33.41 | 38.29    | 37.95 | 33.98 | 35.91   |
| All sites       | Media              | 40.18 | 43.02    | 40.01 | 29.20 | 38.10   |
| Random neighbors | IQR                | 11.77 | 10.55    | 8.56  | 10.59 | 10.37   |

Neighbors: trained using historical data from selected neighbors. This is our target scenario. 2) No Neighbors: trained solely using historical data of the target PV site. 3) All sites: historical data of all PV sites are used for training without detector selection model. 4) Random Neighbors: detector sites are randomly selected.

From Fig. 10 and Table VII, we can see that:
1. When using measurements from only the selected neighbors, all models achieve the best performance (i.e., smallest RMSE median and IQR (Inter-Quantile Range)).

TABLE VIII
FORECASTING PERFORMANCE EVALUATION (AVERAGED ON 95 SITES)

| Scenarios       | Evaluation Metrics | TCN  | CNN-LSTM | VGG-8 | GARNN | Average |
|-----------------|--------------------|------|----------|-------|-------|---------|
| Selected neighbors | RMSE              | 39.80 | 51.88    | 48.52 | 43.81 |         |
| CI-90%           | 10.37             | 15.81 | 16.25    | 10.89 |
| Single site     | RMSE              | 52.86 | 55.80    | 61.77 | 56.71 |         |
| CI-90%           | 11.67             | 18.90 | 15.93    | 12.66 |
| All sites       | RMSE              | 49.92 | 57.74    | 54.30 | 42.15 |         |
| CI-90%           | 17.84             | 23.33 | 25.69    | 14.52 |
| Random neighbors | RMSE              | 54.60 | 52.26    | 58.11 | 49.77 |         |
| CI-90%           | 13.96             | 16.07 | 15.22    | 11.33 |

This demonstrates the efficacy of the neighbor selection method.
1. If measurements from all sites are used, the results have large RMSE variances, showing that data from uncorrelated sites can pollute the forecasting results.
2. Using data from the target site or from randomly selected neighbors can lead to larger forecasting errors, showing the lack of spatial-temporal correlation information.

Next, the four deep-learning models with inputs from selected neighbors are compared for different forecasting horizons. In addition, to compare with non-machine learning methods, we include 2 other baseline methods: persistence model [37] and SARIMA [38]. From results as shown in Figs. 11, 12 and Table VIII, we have the following observations:
1. TCN with detector sites achieves the lowest RMSE, best forecasting stability, and best computation efficiency.
2. Using detector sites mainly improves the forecast accuracy of 2-3 hour-ahead PV forecast. Forecasting errors increase dramatically after hour 4 exceeding that of the physics-based model.
3. We notice that The GARNN model achieves better performance than other methods if all sites are used or if detectors are randomly selected. This is because the attention mechanism in GARNN can dynamically identify the most correlated neighbors and assign them with higher weights, which acts as an “internal neighbor selection” mechanism. However, this also bring additional computing costs.

D. TCN-Based Forecast Reconciliation

After we obtain the forecasting results from both the TF and FF model on their testing set, we align them by time stamps to train and test the reconciliation model (TCN #3). the reconciled forecasting horizon is set to 6h (i.e., the input/output length is 6×12 = 72 in Fig. 4) to reflect the most challenging short-term forecasting scenario. Then we obtain over 400 samples that are evenly distributed from May to October with 6h length. These samples are further divided into training (70%), validation (10%), and testing (20%). The reconciliation model will be trained on the training set, and evaluated on the testing set.
Take site Leo under cloudy/rainy days as an example. As shown in Fig. 13, the FF model uses real-time data as inputs, which cannot capture long-term trends well. Thus, the forecast curve starts to deviate from the actual when the forecasting horizon exceeds 2 hours. This phenomenon can also be observed in Figs. 11 and 12. Thus, without inputs from NWP, FF suffers from large “trend errors”. On the other hand, the NWP based TF forecast, which predicts the long-term trend well, cannot effectively capture intra-hour fluctuations. After reconciliation, the TF and FF results can compensate each other. As the two forecasts are independently generated time series, the reconciled forecasts capture both intra-hour fluctuations and the long-term trend, achieving improved performance.

The reconciliation method is tested on all the 95 PV sites. As summarized in Table IX, after reconciliation, the forecasting RMSE is significantly reduced for forecasting horizon longer than 2 hours, especially under cloudy and rainy weather conditions.

In the end, we give statistics on how many of the 95 PV farms our forecasting framework outperforms the other 3 benchmarking models for 6h ahead forecasting task, shown in Fig. 14. We can see that our method has significant advantages especially when the forecasting horizon is above 2 hours.

### IV. Conclusion

This paper presents a TCN-based hybrid PV forecasting framework for utility-scale PV farms to improve the hours-ahead forecasting performance. In the TF model, TCN is used for NWP input data blending. In the FF model, we develop a scenario-based detector sites selection algorithm that real-time irradiance data from the most contributive neighbors of the target site can be used as inputs to the TCN-based hours-ahead PV forecast model. The TCN hours-ahead model uses spatial-temporal correlation between the detector and target sites to improve the forecast accuracy of large PV output drops at the target site. Then, we develop a forecast reconciliation process to better exploit the capability of the physics-based model for forecasting trend and the capability of the data-driven method for forecasting large cloud events. The results show that the proposed reconciliation model achieve another 30% performance improvement by merging the TF and FF results. We demonstrate that TCN is a comprehensive data-driven solution in solving the short-term PV forecasting problems with superior accuracy and computation efficiency.

Future research will focus on developing a dynamic detector site selection algorithm that can identify the most predictive neighbors in real time considering the upper-air wind direction. Correspondingly, a new forecasting model that can leverage the information of the varying detector sites will be studied.

### APPENDIX

#### A. TCN

In this paper, TCN is used for NWP blending in the TF model, spatial-temporal forecasting in the FF model, and forecasting results reconciliation. The hyper-parameter settings for the 3 TCN models are given in the following Table X.

The key principle of the TCN hyper-parameter selection is to guarantee the receptive field can cover the whole input sequences, according to (2). Besides, more filters mean stronger feature extraction ability. Skip connection is preferred when the network goes deeper to avoid gradient vanishing.

| parameters | NWP blending | FF forecasting | Forecasting reconciliation |
|------------|--------------|----------------|-----------------------------|
| Return sequence | True | False | True |
| Input data length | 24h | 6h | 6h |
| Output length | 1h | 6h | 6h |
| Kernel size | 3 | 2 | 3 |
| Number of filters | 32 | 64 | 32 |
| Dilution rate | [1, 2] | [0, 1, 3, 9] | [0, 1, 3, 9] |
| Number of stacks | 3 | 1 | 1 |
| Skip connection | Yes | No | No |
| Model capacity | ≥ 27K | ≥ 66K | ≥ 26K |

In the end, we give statistics on how many of the 95 PV farms our forecasting framework outperforms the other 3 benchmarking models for 6h ahead forecasting task, shown in Fig. 14. We can see that our method has significant advantages especially when the forecasting horizon is above 2 hours.
issue. We conducted an ablation study for the 3 key hyperparameters that determine the receptive field, i.e., kernel size, dilation rate and number of stacks. Results show that the TCN model performance is stable and is insensitive to different hyper-parameter combinations as long as the receptive field is sufficient to cover the input sequences. However, the model performance will degrade when the receptive field is insufficient. Similar conclusions are also reported in [16].

B. VGG-8

VGG network is a very deep convolutional network proposed by Visual Geometry Group for image recognition purpose [34]. VGG is implemented to the spatial-temporal PV forecasting problem in [35] due to its outstanding feature extraction ability. In this paper we shrink the original 16-layer VGG network to an 8-layer VGG network (i.e., VGG-8) as a benchmarking method for TCN. Model configurations are given in the following Table XI. The model capacity \(\approx 61K\).

C. CNN-LSTM

CNN-LSTM is another popular deep-learning model to solve the spatial-temporal PV forecasting problem [32]. The benchmarking CNN-LSTM model in this paper is given in the following Table XII with the model capacity \(\approx 66K\).

D. GARNN

GARNN model treats the PV sites as a graph, and the correlations among sites are represented by an adjacency matrix. This matrix can be time-varying because the correlation patterns among PV sites are determined by the dynamic weather conditions. To this end, a multi-head attention mechanism is introduced to learn a dynamic adjacency matrix from the time series data of each PV site. Then this dynamic matrix together with the original time series data will be fed into the RNN model to extract the temporal information and achieve forecasting. For more details please refer to [35], [36]. The GARNN model configurations in this paper are given in the following Table XIII. The model capacity \(\approx 64K\).

E. Example of Physics-Based Model Parameters

Table XIV and XV shows the module/inverter parameters of the Leo site as an example.

REFERENCES

[1] J. Antonanzas, N. Osorio, R. Escobar, R. Urraca, F. J. Martinez-de-Pison, and F. Antonanzas-Torres, “Review of photovoltaic power forecasting,” Solar Energy, vol. 136, Oct. 2016, pp. 78–111.
[2] R. Ahmed, V. S. Sreeram, Y. Mishra, and M. D. Arif, “A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization,” Renew Sustain Energy Rev., vol. 124, May 2020, Art. no. 109792.
[3] U. K. Das et al., “Forecasting of photovoltaic power generation and model optimization: A review,” Renew Sustain Energy Rev., vol. 81, pp. 912–928, Jan. 2018.
[4] C. Yang, A. A. Thatte, and L. Xie, “Multitime-scale data-driven spatiotemporal forecast of photovoltaic generation,” IEEE Trans. Sustain. Energy, vol. 6, no. 1, pp. 104–112, Jan. 2015.
[28] R. Levinson, M. Egolf, S. Chen, and P. Berdahl, “Experimental comparison of pyranometer, reflectometer, and spectrophotometer methods for the measurement of roofing product albedo,” Solar Energy, vol. 206, pp. 826–847, Aug. 2020.

[29] W. F. Holmgren, C. W. Hansen, and M. A. Mikofski, “Pvlib python: A python package for modeling solar energy systems,” J. Open Sour. Softw., vol. 3, no. 29, p. 884, 2018.

[30] E. Lorenz, J. Hurka, D. Heinemann, and H. G. Beyer, “Irradiance forecasting for the power prediction of grid-connected photovoltaic systems,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 2, no. 1, pp. 2–10, Mar. 2009.

[31] S. Lu et al., “Machine learning based multi-physical-model blending for enhancing renewable energy forecast-improvement via situation dependent error correction,” in Proc. Eur. Control Conf. (ECC), 2015, pp. 283–290.

[32] H. Zang, L. Liu, L. Sun, L. Cheng, Z. Wei, and G. Sun, “Short-term global horizontal irradiance forecasting based on a hybrid CNN-LSTM model with spatiotemporal correlations,” Renew. Energy, vol. 160, pp. 26–41, Nov. 2020.

[33] H. Wen et al., “Deep learning based multistep solar forecasting for PV ramp-rate control using sky images,” IEEE Trans. Ind. Informat., vol. 17, no. 2, pp. 1397–1406, Feb. 2021.

[34] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, arXiv:1409.1556.

[35] J. Simeunovic, B. Schuhnel, P.-J. Alet, and R. E. Carrillo, “Spatiotemporal graph neural networks for multi-site PV power forecasting,” IEEE Trans. Sustain. Energy, vol. 13, no. 2, pp. 1210–1220, Apr. 2022.

[36] R.-G. Cirstea, C. Guo, and B. Yang, “Graph attention recurrent neural networks for correlated time series forecasting,” in Proc. Model KDD, 2019, pp. 1–6.

[37] S. Dutta et al., “Load and renewable energy forecasting for a microgrid using persistence technique,” Energy Procedia, vol. 143, pp. 617–622, Dec. 2017.

[38] S. I. Vagropoulos, G. I. Choulis, G. E. Kardakos, C. K. Simoglou, and A. G. Bakirtzis, “Comparison of SARIMAX, SARIMA, modified SARIMA and ANN-based models for short-term PV generation forecasting,” in Proc. IEEE Int. Energy Conf. (ENERGYCON), 2016, pp. 1–6.
Si Zhang (Student Member, IEEE) received the B.S. degree in electrical engineering from Wuhan University, Wuhan, China, in 2016, and the M.S. degree from the University of Michigan, Ann Arbor, USA. He is currently pursuing the Ph.D. degree in electrical engineering with North Carolina State University, Raleigh, USA. His research interest includes machine learning application on forecasting, learning-based methods of operation, and control in distribution system.

Abhishek Komandur received the B.S.P.H. and M.S.P.H. degrees in environmental sciences and engineering from UNC Chapel Hill. He is currently a Data Scientist on the Performance Analytics Team with Strata Clean Energy. His work focuses on analyzing data and improving PV performance and forecasting models. His team works on diverse operations activities, including energy forecasting, PV plant performance analyses, PV system performance testing, predictive analytics, building machine learning algorithms, preparing cost-benefit analyses, estimating lost production, cost-modeling, and infrared image analysis.

Laura Kraus, photograph and biography not available at the time of publication.

Taylor Adcox, photograph and biography not available at the time of publication.

Roger Willardson, photograph and biography not available at the time of publication.

Ning Lu (Fellow, IEEE) received the B.S. degree in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 1993, and the M.S. and Ph.D. degrees in electric power engineering from Rensselaer Polytechnic Institute, Troy, NY, USA, in 1999 and 2002, respectively. She is a Professor with the Electrical and Computer Engineering Department, North Carolina State University. Her research interests include modeling and control of distributed energy resources, microgrid energy management, and applying machine learning methods in energy forecasting, modeling, and control.