



Model-driven Per-panel Solar Anomaly Detection for Residential Arrays

MENGHONG FENG, NOMAN BASHIR, PRASHANT SHENOY, DAVID IRWIN, and
BEKA KOSANOVIC, University of Massachusetts, Amherst

There has been significant growth in both utility-scale and residential-scale solar installations in recent years, driven by rapid technology improvements and falling prices. Unlike utility-scale solar farms that are professionally managed and maintained, smaller residential-scale installations often lack sensing and instrumentation for performance monitoring and fault detection. As a result, faults may go undetected for long periods of time, resulting in generation and revenue losses for the homeowner. In this article, we present SunDown, a sensorless approach designed to detect per-panel faults in residential solar arrays. SunDown does not require any new sensors for its fault detection and instead uses a model-driven approach that leverages correlations between the power produced by adjacent panels to detect deviations from expected behavior. SunDown can handle concurrent faults in multiple panels and perform anomaly classification to determine probable causes. Using two years of solar generation data from a real home and a manually generated dataset of multiple solar faults, we show that SunDown has a Mean Absolute Percentage Error of 2.98% when predicting per-panel output. Our results show that SunDown is able to detect and classify faults, including from snow cover, leaves and debris, and electrical failures with 99.13% accuracy, and can detect multiple concurrent faults with 97.2% accuracy.

CCS Concepts: • **Hardware** → **Renewable energy**; • **Computing methodologies** → **Anomaly detection**;

Additional Key Words and Phrases: Solar anomaly detection, data-driven modeling, machine learning

ACM Reference format:

Menghong Feng, Noman Bashir, Prashant Shenoy, David Irwin, and Beka Kosanovic. 2021. Model-driven Per-panel Solar Anomaly Detection for Residential Arrays. *ACM Trans. Cyber-Phys. Syst.* 5, 4, Article 40 (September 2021), 20 pages.

<https://doi.org/10.1145/3460236>

1 INTRODUCTION

Recent technological advances and falling hardware price have led to significant growth in the deployment of renewable solar within the electric grid. The cost of solar deployments have dropped to less than \$2.75 per watt in recent years [2] and have become competitive with traditional energy sources. As a result, utility-scale and residential-scale solar deployments have experienced

This research was supported by NSF grant 1645952, DOE grants DE-EE0007708, DE-EE0008277, and Massachusetts Department of Energy Resources.

Authors' address: M. Feng, N. Bashir, P. Shenoy, D. Irwin, and B. Kosanovic, University of Massachusetts, Amherst, emails: {mfeng, nbashir}@umass.edu, shenoy@cs.umass.edu, irwin@ecs.umass.edu, kosanovic@umass.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2021 Copyright held by the owner/author(s). Publication rights licensed to ACM.

2378-926X/2021/09-ART40 \$15.00

<https://doi.org/10.1145/3460236>

sustained growth across the world, with more than 2.6 GW of deployments in 2019 Q3 in the U.S. alone [2].

Typically, larger utility-scale solar farms are professionally monitored and maintained for optimal performance—they are instrumented for monitoring real-time generation to identify production issues, and also cleaned frequently to reduce dust or pollen. Researchers have also suggested using drones carrying thermal cameras to identify and locate faults in large solar arrays [8]. However, the majority of solar installations today are small-scale installations, often on residential rooftops, with capacities of less than 10 kW in 2018 [1]. Due to cost reasons, such systems lack sensing and instrumentation that may be present in larger utility-scale solar farms. Further, monitoring of these systems is left to homeowners, who lack the technical expertise for this task. At best, system performance may be monitored at a coarse-grain systemwide basis to determine system-level issues. As a result, it is not uncommon for residential solar arrays to encounter power anomalies or other local faults that go undetected for long periods of time, resulting in a loss of generation and revenue for the owner. While it is possible to add sensors and instrumentation for real-time monitoring, doing so for small-scale installations increases their cost, and is challenging to do for millions of installations that are already operational without such capabilities.

In addition to initially detecting an anomaly, classifying the type of anomaly is important for several reasons. First, some anomalies “go away” after a few days, e.g., snow that melts or slides off, and do not need any action. In contrast, some anomalies “persist” for days, e.g., electrical fault or wet dust/leaves, and require immediate action. Second, manual inspection is quite labor-intensive and expensive, possibly requiring travel to the sites that are remote, which is especially true for utility-scale solar sites that typically do not have on-site maintenance. Third, classification shows the difference between shading and other types of issues, which is useful. If fault is small compared to a site’s size, then the operator may decide to postpone an inspection until next regular inspection.

To address these challenges, we present *SunDown*, a sensorless approach for detecting and classifying per-panel faults in small-scale solar arrays. Prior work on per-panel solar anomaly detection are based on time-series [19] or statistical [5, 35] analysis of a panel’s output or use of sensors such as a pyranometer [16] to detect faults. In contrast, our approach uses the actual output from *other* nearby panels to estimate each panel’s expected output and find anomalous deviations from this estimate. *SunDown* assumes that per-panel generation information is available from the array—an assumption that holds true for any installation that uses micro-inverters or DC power optimizers—and uses a model-driven approach to detect when the panel output deviates in an anomalous manner from the model-predicted output. Our approach is based on machine learning and can detect physical anomalies such as snow obstructions, leaves, and electric faults at panels. Our approach seeks to identify and alert solar owners of such issues in a timely manner so that they can be rectified to avoid production losses.

In designing, implementing, and evaluating *SunDown* we make the following contributions.

- We present a model-driven approach, based on machine learning, that leverages correlations in the generated output between adjacent panels to predict the expected output of a particular panel and flags anomalies when the model predictions deviate from the expected values. Further, our approach can handle and detect multiple concurrent faults in the system.
- We present a random forest-based classification technique to classify the probable cause of the observed fault. To validate our approach, we construct two labelled datasets of solar anomalies: a two year dataset from a real-home with real snow cover anomalies that we hand label using ground-truth information and a solar anomaly dataset that we construct with a 20-panel array by injecting synthetic faults such as dust, leaves, and open circuit faults.



Fig. 1. A residential solar array (right) with 31 panels deployed on four roof planes and real-time panel-level generation data from the array (left).

Since there is a dearth of solar anomaly datasets, we release both datasets and our code as open-source tools to the community.

- We conduct a detailed experimental evaluation of our methods. We show that our approach has a **Mean Absolute Percentage Error (MAPE)** of 2.98% when predicting per-panel output, which shows the efficacy of using neighboring panels to perform model-driven predictions. Our results also show that SunDown is able to detect and classify faults such as snow cover, leaves, and electrical failures with 99.13% accuracy for single faults and is able to handle concurrent faults in multiple panels with 97.2% accuracy. Our results also demonstrate that the approach is robust to the solar site’s asymmetries (different orientation, different shading), as the solar site used in experiments has many asymmetries (as shown in Figure 1).

2 BACKGROUND

In this section, we present background on residential solar arrays and solar anomaly detection.

2.1 Residential Solar Arrays

Our work primarily focuses on residential solar arrays, such as ones often found on residential rooftops. Such installations are typically small-scale installations with capacities of 10 kW or less and comprise a few to a few dozen solar panels (see Figure 1). Since we are interested in monitoring anomalies and faults at a per-panel level, we assume that the power generation of the array can be monitored at a per-panel level.

This is a reasonable assumption in practice, since many residential arrays are equipped with micro-inverters (e.g., Enphase micro-inverters [3]) or DC power optimizers [4] on each panel that are designed to track and independently optimize the power generation of each individual panel. Such installations, which are now commonplace, are advantageous, since they maximize the total system output even for deployments that span multiple roof surfaces and under partial shading-effects. As shown in Figure 1, such systems provide real-time per-panel generation data, which is essential for our approach. Other than knowledge of per-panel output, we do not assume any other sensors or instrumentation on the residential solar installation. Thus, we seek to develop a sensor-less approach for per-panel solar anomaly detection.

2.2 Solar Generation

It is well-known that solar generation at any site depends directly on the amount of sunlight—solar irradiance—received at that location. The solar irradiance is a function of the latitude and longitude of that location and the season of the year [22]. Of course, the weather—specifically cloud cover—can reduce the solar irradiance at a particular site. For the purpose of this work, we assume that per-panel solar generation on any given day can be reduced to two factors: *transient*,

which comprises of factors that temporarily impact power output, and *faults* that comprise factors that have a prolonged negative impact on output.

Transient factors include weather conditions such as cloud cover, wet panels caused by rain or dew, as well as site specific factors such as shading caused by nearby trees or other structures. We can classify transient factors into two classes—common or local. Common transient factors are those that impact all panels of a particular site such as overcast condition or rain. Local factors are those that impact a particular panel, or a group of panels, but not all of the panels at that site. For example, many shading effects may impact a portion of the site, depending on the foliage and the location of the sun.

2.3 Solar Faults

Anomalies (also referred to as faults) in our case are defined to be factors that cause a persistent drop in production but can be rectified by the owner of the site. We are particularly interested in the following three types of faults (1) snow cover on one or more panels, (2) partial occlusions such as bird droppings, dust or leaves on a panel, (3) electric faults such as module failure, short circuits or open circuits. These faults cause either a reduction in output or zero output for a particular panel or a subset of panels.

Due to their close proximity to one another, multiple panels in a residential array may experience the same fault—for example, snow may cover multiple adjacent panels (or even the entire system), resulting in concurrent faults. Of course, a site may also suffer a full system outage, which is also a fault but is easier to detect than those that cause partial outages or partial output reduction.

2.4 Other Sensorless Approaches

There have been other sensorless approaches for solar fault identification, both at a system and per-panel level. There is prior work on solar fault detection and analysis using drones and computer vision, which does not require instrumentation on the solar site [6, 23]. There is also prior work that uses historical data to model the power output of the panel or system under different conditions and uses this model to predict the current output and compare it to observed values. For instance, such approaches have been based on time-series [19] or statistical [5, 35] analysis of a panel's output or use of sensors such as a pyranometer [16] to detect faults. However, historical data dependency prevents using these approaches for newer installations until enough history (6+ months) is available. Spatial methods do not need as much temporal history, since they exploit spatial relationships. Additionally, even state-of-the-art models that use historical data for predicting future solar power generation, e.g., hybrid physical and ML models, have an $\approx 20\%$ MAPE when trained and tested at the same solar site [11]. Such a large error is not suitable for detecting and classifying anomalies that experience small deviations. As a result, rather than using *temporal* correlations in output on a particular panel, our approach uses *spatial* correlations that exist between outputs of neighboring panels, a complementary approach that enables a model to be trained for a site without requiring a lot of historical data (e.g., new sites).

2.5 Problem Statement

Consider a solar array with N solar panels. We assume that the panels are mounted on a residential roof and may be mounted on one or multiple roof planes. Note that in the latter case, panels will have different tilts and orientations. We assume that the power generated by each panel can be monitored in real time and that the weather at the site is also known (e.g., from a weather service). Given such a setup, our problem is to design a technique that monitors the power output of each panel and the entire system, and labels the observed output in each time interval (e.g., a day) as normal or abnormal. Further, our technique should identify specific solar panels in the system that

are experiencing faults and also determine possible cause of the fault (e.g., snow, partial occlusion, or electric fault).

3 PER-PANEL SOLAR ANOMALY DETECTION

In this section, we describe our model-driven approach for per-panel solar fault detection and how we can build on this approach to perform multiple fault detection. We first describe the basic idea, followed by the details of our models and algorithms.

3.1 Basic Idea

Consider a solar installation with N panels. Suppose that k panels are experiencing an anomaly that result in a reduction, or loss, of output from those panels. Initially, let us assume $k = 1$ (only one panel out of N is faulty). Later, we show how our approach can be extended to handle multiple, concurrent faults where $k > 1$.

Since all N panels are mounted on the same roof in close proximity of each other, it follows that they experience highly correlated weather conditions, and produce similar output. Thus, our “sensorless” approach first constructs a model to predict the expected output of a panel from n neighboring panels ($n \geq 1$). For example, a simple predictor is one that uses the mean output of n neighboring panels to estimate a particular panel’s output. Under normal conditions, since adjacent panel outputs are highly correlated, the model prediction will match the observed output of that panel with high accuracy. Note that any n out of the available N panels can be chosen to model the output of a particular panel. A useful heuristic is to use the “closest” n panels to the one being predicted or to use the n panels on the same roof plane, since they will have higher correlations than those on a different roof surface of the same house. In our evaluation, we experimentally evaluate the accuracy of these heuristics and also evaluate the value of n that yields sufficient accuracy.

When a panel experiences an anomaly, however, the model predictions will continue to estimate the “normal case” output of that panel, while the observed output will deviate from this normal case. If the deviation is “large” and persists over an extended period of time, then it is indicative of a fault, rather than an error in the model prediction. The quantitative value of “large,” in terms of expected power deviation, depends on the expected error in the predictor used to get the panel’s expected *normal* output. The deviation should be greater than the expected error in the predictor. Based on the expected error, the threshold for an anomaly can be set to a value $>(\text{expected error})$. Any observed deviations greater than the expected error are then labeled as an anomaly. The amount of time the deviation should persist is configured to avoid shading or similar effects from raising false positives. In general, an anomaly should persist for a period of days, rather than minutes or hours. As we show in Section 4, the right number of days can be empirically determined using analysis of historical anomalies data. The cause of the fault can be separately determined by analyzing amount of loss or the power pattern exhibited by the panel. Such a model-driven approach only uses the observed output of panels to detect anomalies—no other instruments or sensors are needed for anomaly detection unlike some other approaches [8].

3.2 Model-based Predictions

Based on the above intuition, we now present two model-driven techniques for predicting the power output of an individual panel using neighboring panels. Our first model is based on linear regression and uses only power output of panels as input parameters to make predictions. Our second model is based on a probabilistic graphical model and half-sibling regression.

3.2.1 Linear Regression-based Model. Since the power generated by solar panels in close proximity of one another are highly correlated, we can use regression to predict the output of a panel given the observed output of neighboring panels.

Let P_i denote the observed power output of panel i at time instant i . Let us assume we wish to predict the output of panel i using n other panels. Typically, we can choose n nearest panels, or n panels on the same roof plane, from the N total panels on the roof. A linear regression model allows us to estimate the output of desired panel as a linear function of the others:

$$P_i = w_1 P_{i1} + w_2 P_{i2} + w_3 P_{i3} + \cdots + w_n P_{in} + \epsilon_i, \quad (1)$$

where $X = \{i_1, i_2, \dots, i_n\}$ is the set of n panels used to model the output of the i th panel. We can use linear regression to estimate the weight w_i that minimize the error term ϵ_i .

Such an approach yields N distinct regression models, one for each panel in the system, where each model makes prediction using the observed output of n other panels. To determine if a panel has a fault, we compare the model predictions at time t , $P_i(t)$ with the observed value $\hat{P}$. If the difference between the model predictions and observed values is large and persists over a period of time (e.g., a day or multiple days), then the approach flags that panel as faulty.

3.2.2 Graphical Model and Half-Sibling Regression. Our second model is based on a recently proposed machine learning technique called half-sibling regression that uses a Bayesian approach to remove the effects of confounding variables [33]. This approach has been used by astronomers to remove noise from measurements of multiple telescopes observing the same phenomena. The main intuition behind the approach can be understood from the astronomy use-case. Suppose that $n + 1$ telescopes are observing the same object such as star. The observations will have some “common” noise introduced by factors such as air pollution or haze that impact visibility of the object. Furthermore, each telescope will have local factors such as instrument calibration error that introduce additional local errors. If we use observations of n telescopes to estimate the expected observation of the $(n + 1)$ -st instrument, and take the difference between the observed and predicted values, then we are left with the local errors (“anomalies”) at that instrument. In our case, we have $n + 1$ solar panels “observing” the sun—their power output represent their observations of the sun. All panels see common factors such as clouds that introduce similar output reductions in the power values. Further, each panel has local factors such as shade (transient factors) or faults that can result in additional reductions in the power output. If we use n panels to predict the output of the $n + 1$ -st panel using a Bayesian model, then the difference between the predictions and observed output should isolate local factors including the effect of faults. This is the intuition behind using the Bayesian approach in Reference [33].

More recently, this approach was used in a system called SolarClique [21] to predict the output of an entire array using nearby solar arrays. We draw inspiration from the half-sibling regression paper [33] and SolarClique [21] for SunDown’s anomaly detection, but point out important differences between the SolarClique method and our approach as shown in Table 1. First, SolarClique is designed for system-level predictions (predicting the total generation of an entire array) and does not have the capability of making fine-grain per-panel predictions, which is the focus of our method. Second, a key technical limitation of SolarClique is that it assumes a single fault can occur at a time, and that the system is not capable of scenarios where multiple arrays are faulty. This is a reasonable assumption for SolarClique, since it uses n arrays from n different homes to predict the output of a specific home, and faults across arrays and homes can be assumed to occur independently. In our case, since panels are in close proximity to one another, the same fault (e.g., snow) can impact multiple panels, and faults therefore no longer occur independently. Since the independence assumption of SolarClique does not hold in our case, a key technical improvement over prior work is our ability to handle multiple faults (as discussed in the next section). For simplicity, we first assume a single fault in the entire system and present our approach. We then relax the assumption in the next section and show how the basic model can be extended to handle multiple

Table 1. A Comparison of the State-of-the-art SolarClique System and Our SunDown Approach

	SolarClique	SunDown
Per-Panel faults	No	Yes
Systemwide faults	Yes	Yes
Multiple faults	No	Yes
Anomalies Detected	Systemwide electrical	Snow, electrical, occlusion

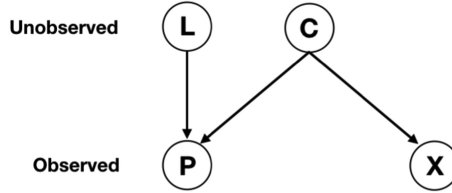


Fig. 2. Graphical model representation.

concurrent faults. A final difference is that SolarClique did not focus on fault classification (and only detects large system-level electrical failures) while SunDown can identify multiple types of faults, including snow cover, occlusion faults and electrical faults.

To describe our Bayesian model, let P be a random variable denoting the power output of a particular panel. Let X denote a random variable representing the power output of n other panels in the system. Hence, X is a vector of size n . Let C denote the confounding variables that impact both X and P . In our case, C denotes common confounding variables such as cloud cover that have the “same” impact on panels. Let L denote the local factors and asymmetries between panels that impact the output of an individual panel. L will include transient factors, including partial shading, as anomalies that locally impact P . The relationship between P , X , L , and C can be captured using a (causal) graphical model as shown in Figure 2. Since the output of each panel can be directly monitored, P and X are observed variables, while C and L are latent unobserved variables.

As can be seen, P depends on both L and C while X depends only on C (and is independent of L). C impacts X , and when conditioned on P , P becomes a “collider,” making X and L dependent. To reconstruct L using half-sibling regression, we assume the following additive model:

$$P = L + f(C). \quad (2)$$

Since C is unobserved, we can use X (which is observed) to approximate $f(C)$. If X exactly approximate the function $f(C)$, then we can then compute $f(C)$ on $E[f(C)|X]$. Even otherwise, if X is a sufficiently large vector, then it can yield a ground approximation. Thus, we can use X to predict P and recover L from Equation (2) as

$$\hat{L} = P - E[P|X]. \quad (3)$$

Note that $\hat{L}$ estimates both anomalies and transient factors, and the impact of transient factors must be removed from L to estimate the anomaly.

Given these concepts, our algorithm to estimate the amount of production loss due to anomalies is as follows.

We first use regression to estimate P using X . This is similar to the linear regression method from the prior section. The regression yields $E[P|X]$ —an estimate of P given the observed output of n neighboring panels that constitute X . Since P itself is observed, subtracting $E[P|X]$ from P yields

an estimate of the output loss $\hat{L}$ due to transient factor and anomalies as shown in Equation (3). A key difference between linear regression model of Section 3.2.1 and here we use bootstrapping to construct multiple regression model by subsampling the data (instead of a single regression model) and use an ensemble method based on Random Forest that uses the mean of multiple models to estimate $E[P|X]$.

Next, since $\hat{L}$ contains effects of transient factors such as shade on panels as well anomalies, we must remove the impact of transient factors to obtain the “true” anomalies. We can use time-series decomposition to extract the seasonal component that represents shading effect that occur daily at set time periods and remove it from $\hat{L}$ [21]. The remainder of $\hat{L}$ represents production loss at that panel due to any anomalies. Under normal operation $\hat{L}$ will be close to zero (no anomalies and no loss of output). When $\hat{L}$ is significant and persistent over a period of time, our model-driven approach flags an anomaly in the panel.

3.3 Handling Multiple Concurrent Faults

Both our regression and Bayesian models use the power output of n panels to predict the expected output of another panel. A very important assumption is that the n panels being used as inputs to the model are non-faulty and hence be used to predict the normal case output of another panel. An anomaly is flagged when the model prediction of normal case output deviates from the observed output, indicating the presence of an anomaly.

This approach works well when there is only one faulty panel in the system—which implicitly implies that all remaining panels are non-faulty and any model that uses some of these remaining panels to make predictions will have “clean” non-faulty inputs. However, due to the close proximity of panels, anomalies such as snow cover, dust, leaves, are likely to impact multiple panels. In this case, some of the inputs to the model may come from faulty panels, causing model prediction to have high errors. Of course, if n is made large and only a small number of panels are faulty, then the model may be able to tolerate the “noise” in a small number of inputs and still produce reasonable accurate prediction. However, many residential rooftops may have a small number of panels, which means n can not always be large. Hence, we need an explicit method to tolerate the impact of multiple concurrent faults in the system.

Observe that our models use *any* n out of N total panels to predict the output of panel i . Thus, it is possible to construct *multiple models for each panel* by choosing different subsets of n panels out of N , and then using them as inputs to predict the output of panel i . In the normal case (no faults), all of these models show similar predictions for panel i ’s output. However, when multiple panels are faulty, any model that uses faulty panels as input will have higher errors while a model that uses all non-faulty inputs will continue to provide good predictions. Our goal then is to construct multiple models for each panel using our Bayesian or regression method, and then choose one of these models at each instant that uses non-faulty inputs.

To do so, we need to distinguish between faulty and non-faulty inputs. However, since the models are themselves being used to detect faults, we need a different method to determine which inputs are possibly faulty. To do so, we use a solar forecasting approach that predicts the output of the solar panel based on weather forecasts. There is extensive work on solar forecasting using weather forecasts and any such model can serve our purpose. We use a machine learning forecasting-based model that uses the location of the system (longitude and latitude), time of day, past power observations and near-term weather forecasts (e.g., sunny, cloudy) to estimate the output of a panel [22]. This model, and many others, have been implemented into the Solar-TK open-source library [9], which we leverage to design a custom forecasting model for each panel in the system using near-term future weather forecasts.

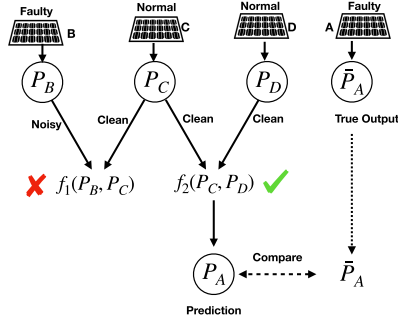


Fig. 3. A forecasting model is used to ensure non-noisy inputs to our Bayesian model.

Suppose that $P_i(t)$ is the estimate of power output of a panel i based on this forecasting model. If $P_i(t) - \bar{P}_i$ is large, then it implies that expected output differs from the prediction and the panel is possibly a “noisy” input. Our per-panel forecasting models perform these prediction for each panels and labels it as “noisy input” or “normal input.” Any model that uses one or more noisy panel as an input should be eliminated from consideration for anomaly detection purposes. That is, SunDown chooses any regression or Bayesian model (out of multiple models for a panel constructed from different subsets comprising n panels) such that all inputs to that model are labelled normal.

Consider the following example to illustrate the process (Figure 3). Suppose that a solar rooftop install has four panels: A, B, C, D . We wish to predict the output of panel A using two other panels. Suppose both A and B are faulty. Let us assume we have the following two half-sibling regression-based Bayesian models, f_1 and f_2 to predict P_A , the power output of panel A ,

$$P_A = f_1(P_B, P_C), \quad P_A = f_2(P_C, P_D), \quad (4)$$

where model f_1 predicts A using panels B and C as inputs, while f_2 predicts A using C and D . Our approach first predicts P_A, P_B, P_C , and P_D using per-panel machine learning solar forecasting models for each of the four panels [9]. Since A and B are faulty, they get labeled noisy inputs. Hence, f_1 is eliminated from consideration, since one of its inputs, P_B , is a noisy input and f_2 is chosen for prediction, since both its inputs, P_C and P_D , are labelled “normal.” Using model f_2 yields a better estimate for P_A than model f_1 . Note that, doing so enables us to handle concurrent faults—we can avoid using faulty panels as model inputs, and at the same time, use our Bayesian method to identify the presence of multiple faults.

We note that our solar forecasting models are only used to identify noisy model *inputs* rather than to determine anomalous solar *output*—although our solar forecasting models also provide an estimate of the panel’s output, they are not suitable for anomaly detection. This is because they use weather forecasts of cloud cover, along with other parameters, to estimate a panel’s output. Forecasts of future weather are inherently error-prone, which means the forecasting model will also have higher errors. Using a solar forecasting model directly for anomaly detection will also have higher false positive (due to model errors). A higher error is acceptable when detecting noise in input data, since noisy inputs due to anomalies will cause “large” deviations and are simple to detect. In contrast, the Bayesian approach uses actual power output observations to estimate a panel’s output for the purpose of anomaly detection, which yields a more accurate model and reduces changes of false positives. As shown in Section 5.1.1, our model has less than 3% error. Consequently, we use forecasting models to only identify noisy inputs and eliminate Bayesian models with noisy inputs as shown in Figure 3; incorrectly labeling a panel as noisy due to forecasting

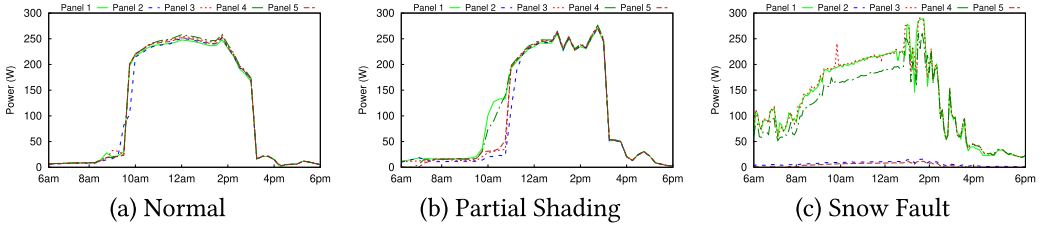


Fig. 4. Residential home power output on an example day under (a) normal condition, (b) partial shading on some panels on east side, and (c) snow covering on some of the panels.

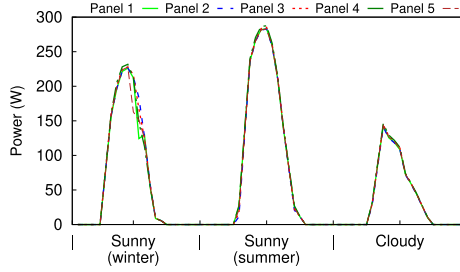


Fig. 5. Panel output on sunny day in summer and winter and a cloud day.

error only causes some of the models to be suppressed for anomaly detection and does not impact accuracy of the remaining models for finding faulty panels.

4 CLASSIFYING SOLAR ANOMALIES

While the previous section presented model-driven approaches to detect the presence of anomalies in one or more panels, in this section, we present a classification approach to determine the possible causes of the output loss seen at the panel(s).

4.1 Solar Anomaly Open Dataset

To assign a possible cause to an observed output loss, we must analyze the observed power pattern and match it to the “power signature” exhibited by different type of solar faults. However, this requires that we have ground-truth data for various type of faults, which is challenging, since there are no open datasets of solar faults available for research use (solar farm operators likely have such data but have not released it to others). Consequently, we need to gather our own data with ground-truth information on solar faults.

Our anomaly dataset contains data from two residential scale solar installations:

- (1) a 31-panel, 9-kW solar installation (Figure 1, top) that experienced multiple snow cover anomalies (Figure 1, bottom) over its two year lifetime
- (2) a 20-panel ground mounted solar installation (Figure 7) where we systematically introduce anomalies such as dust, leaves, electrical faults, and so on, to mimic real-world faults and measure its impact on the output.

We discuss each dataset in more detail before describing our classification method.

4.1.1 Snow Anomaly Dataset. This dataset comes from a residential solar array deployed on a home in Northern America (location details removed for double blind renewing). The house contains 31 rooftop panels, mounted on four different roof planes, as shown in Figure 1 (bottom).

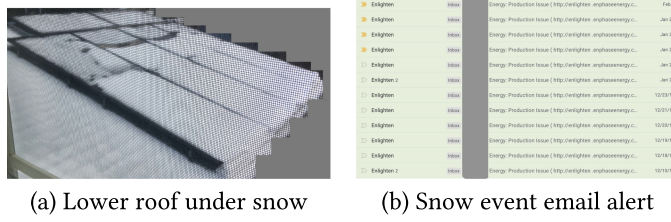


Fig. 6. Snow anomaly dataset.

Each panel is a 320-W LG panel with an Enphase micro-inverter that can optimize the panel's output independently of the rest. As noted earlier, micro-inverters optimize and report panel-level generation data, which is a prerequisite for our models.

We have been gathering data from this system for over 2 years and have per-panel generation information at 5-minute granularity from September 2017 to February 2020. We have also gathered weather data for the location from DarkSky and NOAA weather service.

The only real anomaly encountered by this system over the two-year period is snow cover, following a snow storm (the area receives frequent snowfall in the winter). Depending on how long the snow sticks on the panels following a snow event, snow-covered panels may produce little or no output. As snow melts, some panels generate output, while others stay covered with snow (Figures 4(c) and 6(a)).

We have two sources of ground truth to label snow faults. First, the Enphase system sends an email to the homeowner when it observes near zero output for an entire day, as shown in Figure 6(b). The email indicates a "possible production" issue at the system. Second, DarkSky and NOAA provide past weather data, such as snow events and the extent of the snowfall at a location.

We use both sources of information (which match closely with each other) to manually inspect the per-panel generation data on a snow day and the following several days. We then hand label each panel's output as normal (if it produces any output) or as a snow anomaly (if the panel output is near zero). This yields a hand-labelled dataset of snow anomalies.

4.1.2 Solar Anomaly Dataset. Using our 20-panel ground mounted experimental array and sensors to measure its power output, we carefully introduced several types of anomalies onto specific panels, and measured its impact on the power output. We conduct several data gathering experiments over a period of several weeks under different conditions (sunny, partially overcast, overcast, etc.) and gathered data for the following anomalies.

- (1) Leaf occlusion: We introduced different number leaves on panels (partial occlusion anomaly) and measured its impact
- (2) Dust occlusion: We added different amounts of dust on the panels and measured its impact
- (3) Water drops occlusion: We add varying amount of water drops on the panel and measure its impact. This is designed to mimic morning dew on panels, which is not a true anomaly but a weather effect
- (4) Open circuit fault: We used a variable potentiometer to introduce a high resistance seen by the panel to mimic an open circuit fault and measured its impact.

This hand-crafted anomaly dataset, along with photographs and labels, provides an additional source of data for our experiments. For example, Figure 7 shows leaves on the panel that emulate a partial occlusion fault. Figure 4(a) and (c) depicts the output of the panels in normal conditions and under a snow fault, respectively. Figure 10(a) and (b) illustrates the power output under



Fig. 7. A synthetic leaf occlusion fault in our experimental array.

synthetically generated open circuit fault and a partial occlusion fault. We have released both datasets to the research community.

4.2 Classifying Anomalies

Given anomalies detected by our Bayesian model we use a random forest classifier to label the possible cause of the fault for each panel that is faulty. The classifier needs to distinguish between three types of faults: snow, partial occlusion, and open circuit. Note that partial snow over on a panel and partial occlusion faults both result in diminished but non-zero output. Full snow cover on a panel and open circuit faults both yield zero output. To distinguish between these cases, we first sample 40 randomly chosen points over an entire day and compute the percentage reduction in power output when compared to the model predictions for each of these points. This power loss vector is a key feature to our classifier. We also use two other features: month of the year and snow depth values from NOAA weather service. We train our random forest classifier using a training dataset of real snow and synthetic anomalies. Depending on the season (winter versus other seasons) and the observed power loss over a period of time, our classifier can label the probable cause of fault for each panel. In the current set of experiments, if the deviation greater than the specified threshold persist for two days, then we classify the deviation as the anomaly. Our approach can also label systemwide faults, caused either by a systemwide electrical failure or full snow cover on the entire system, both of which cause near total loss of power output.

5 EXPERIMENTAL EVALUATION

We evaluate SunDown by quantifying (1) the accuracy of model-based power inference where we infer the output of a single panel using nearby panels; (2) the impact of parameters such as number of panels, roof geometry, and weather; and (3) the accuracy of our anomaly classification. We quantify the accuracy of predicting a panel's output using MAPE between the inferred output and the actual solar generation, as follows:

$$MAPE = \frac{1}{m} \sum_{t=1}^m \left| \frac{P_O(t) - P_I(t)}{\bar{P}_O} \right|, \quad (5)$$

where m is the number of samples, $P_O(t)$ is the observed solar power at time t , $P_I(t)$ is the inferred power at time t , and $\bar{P}_O$ is the mean of observed power generation. Equation (5) is an alternative form of standard MAPE where we replace the denominator comprising a single observed value by the mean of all observed values. The alternative form avoid divide by zero issues when the denominator (and observed value) are zero. For the anomaly detection and classification tasks, our goal is to correctly classify all the different anomalies. We use four different metrics to quantify different aspects of the classification task: accuracy, precision, recall, and F1 score. The accuracy is computed by dividing the number of correctly classified anomalies by the total number of anomalies. Precision quantifies what proportion of predicted anomalies are true anomalies. Recall quantifies

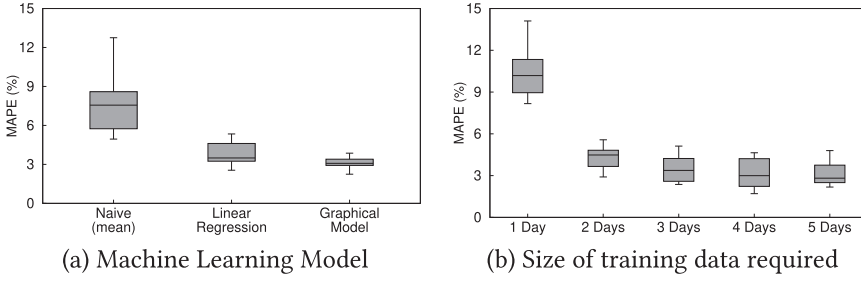


Fig. 8. Prediction model accuracy and the size of training data required.

what proportion of actual anomalies are predicted as anomalies. Finally, F1 score presents a balance between precision and recall: If either precision or recall is low, then F1 score will be low. The different metrics are computed as follows:

$$Accuracy = \frac{TP + TN}{N}, \quad Precision = \frac{TP}{TP + FP}, \quad (6)$$

$$Recall = \frac{TP}{TP + FN}, \quad F1 \text{ score} = \frac{2 \times Precision \times Recall}{Precision + Recall}, \quad (7)$$

where N is the total number of instances, TP is the number of anomalies correctly classified, TN is the number of normal days correctly classified, FP is the number of normal days classified as anomalies, and FN is the number of anomalies misclassified as normal days. Accuracy is used to evaluate the overall model's performance, while sensitivity and specificity are used to test how accurate the model is to correctly detect the anomalies and normal cases.

5.1 Prediction Model Accuracy

We begin by evaluating the accuracy of predicting the power output of an individual panel using neighboring panels.

5.1.1 Machine Learning Model. To evaluate the accuracy of model inference, we choose a test data only from the days where the site experiences no anomaly. We then use the normal days of the home dataset to train our linear regression and graphical model. We also compare their performance with a naive approach that infers the power output of a panel as the mean output of n other panels. We then compare the model predictions using a test dataset and compute the MAPE values for each approach. As shown in Figure 8(a), the MAPE values for Bayesian model, linear regression, and naive approach are 3%, 4%, and 8.6%, respectively. The naive approach has the worst accuracy, since it all panels produce similar output, which is not true in many cases due to panel-level variations. Linear regression works well when the output of different panels are highly correlated and have a linear relation between them, which is not true when some of the panels experience partially shading. Our graphical ensemble learning approach is able to model non-linear relationships and yields highest accuracy and a tight confidence interval. We use the graphical model for the subsequent experiments, unless stated otherwise.

5.1.2 Impact of Training Data Size. Next, we evaluate model accuracy for different amounts of training data. If a model requires a lot of training data for good accuracy, then it can hinder its use for recently deployed solar sites or for the sites where long-term panel-level data are not available. We vary the training data size (by randomly choosing a certain number of days) and evaluate its accuracy for predicting output using a test dataset. Figure 8(b) demonstrates that our model can

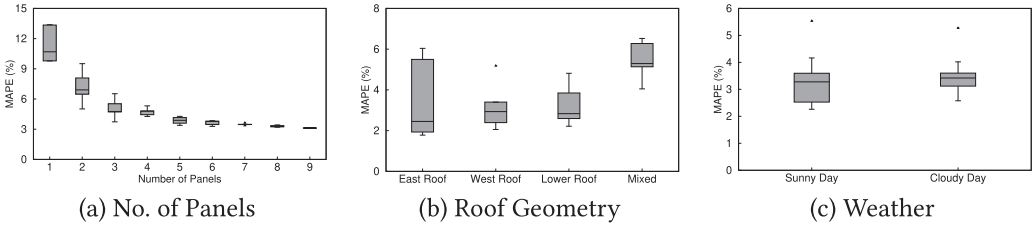


Fig. 9. Effect of various factors on the model accuracy (a) number of panels, (b) roof geometry, and (c) weather.

achieve a decent accuracy and a 10% MAPE with only one day of per-panel data. If the number of days is increased to 4, then the MAPE drops to 3.5% and stays almost constant beyond four days.

Results. Our graphical model can predict per-panel output with 2.98% MAPE and outperforms linear regression and a naive averaging approach. The random forest-based ensemble graphical model does a better job of capturing non-linear relationships among less-correlated data than linear regression. While model accuracy increases with training data size, even only four days of training data yield good accuracy. This result also informs our selection of the threshold for power deviation. The expected (mean) error for our graphical model is 2.98%. However, since the maximum error we observe for our model is $\approx 5\%$; we set the anomaly threshold to be $>5\%$; any deviation greater than $>5\%$ is labeled as an anomaly.

5.2 Impact of Parameters

We next investigate various factors that impact the inference accuracy, including number of panels, geometry of the solar deployment and weather.

5.2.1 Impact of Number of Panels. The individual solar panels at a site can demonstrate subtle variations in their solar output, despite their close proximity, due to panel-level dust, different tilt and orientation angles, and panel-level physical faults such as cracked glass. To evaluate how many panels are need by a model to provide adequate accuracy, we vary n (the number of panels used by the model as input) and compute MAPE for different n . Figure 9 shows inaccuracy is high when using less than 3 panels for inference. The accuracy improves as number of panels is increased to 5 and shows diminishing gains beyond that. The model has an average MAPE value of only 3–4% and a very tight bound, when using 5 panels, as compared to 9% MAPE with single panel. This result suggests that SunDown requires as little as 5 panels to be highly accurate. A typical residential site consists of 2–10 kW installations and contains 6–30 solar panels. As SunDown requires only around 5 panels while yielding high accuracy, it is applicable to all types of solar sites, residential or otherwise.

5.2.2 Roof Geometry Impact. The output of a solar panel depends upon its tilt and orientation, among other factors [12]. Since a residential array may be installed on multiple roof planes, it is preferable to use panels on the same roof plane to predict others, since they will have a similar tilt and orientation and will exhibit higher correlations.

To evaluate the effect of roof geometry, we split the home dataset into four sub-datasets based on the four roof planes where the panels are deployed. We created four graphical models to predict the power output of i th panel by using $n = 7$ panels as inputs. For the east roof, west roof, and lower roof cases, all seven input panels are mounted side by side on the same roof plane facing the same direction. In the fourth scenario, we created a mixed dataset by combining two panels from each east roof and west roof datasets, and three panels from the lower roof dataset. Figure 9(b)

illustrates the inference accuracy as the geometry of panels used for inference varies. For the same roof plane, the model is highly accurate and the MAPE value is between 3% to 3.2%. The large variation for the east roof is due to the partial shading on some of the panels on the roof, leading to inaccurate inferences. To isolate the effect of shading, we used Solar-TK's shade-adjustment module that provides a shading factor, as a value between 0 and 1, that indicates the level of shading on a solar site, where 0 means no shading and 1 means fully shaded. We adjust the output of our shaded panels by this factor to get the expected power without shading. We ran the experiments both with and without shading filtered out, and got the same results. This results illustrates that (1) our approach can filter shading out if necessary, and (2) that in this case, our approach is robust to any shading related anomalies, which tend to be small and transitory. The average MAPE of 5.5% for the mixed dataset demonstrates that our model produces a reasonable accuracy even when input panels are chosen from different roof planes. Thus, when knowledge of the roof geometry is available, it should be exploited, but the model works well even for systems where the roof geometry may be unknown causing the model to use panels from different roof planes for inference.

5.2.3 Impact of Weather. The weather at a solar site, primarily cloud cover, impacts the power generation of a site. On a sunny day, all the solar panels produce similar amount of power. However, on a cloudy day, scattered clouds may only cover one or few of the panels leading to power variation across panels, which can complicate inference. Figure 9(c) illustrates the effect of weather on the accuracy of the inference task. Our model achieves similar mean accuracy on both sunny and cloudy days, indicating it performs well regardless of weather. The higher variance in MAPE on a sunny day is due to shading from nearby structures, and that has a more prominent impact on a sunny day over a cloudy one.

Results. Our experiments show that the number of panels used for prediction as well as the roof geometry play an important role in the model's performance. We find that model yields higher accuracy when five or more panels are used for predictions and when these panels are co-located on the same roof plane. The weather conditions, however, do not impact model accuracy.

5.3 Anomaly Classification Accuracy

The previous section evaluated the accuracy of our model in predicting the output of a panel using nearby panels. We next evaluate the accuracy of our model-driven approach and its classifier in detecting and classifying anomalies, respectively. We compare the accuracy of our approach against a baseline that uses a simple linear regression model to predict the panel-level output and detect anomalies. The linear regression model is the same as shown in Figure 9. The classification aspect of the linear regression-based approach are the same as our bayesian approach: Deviations greater than 5% that persist for at least two days are considered anomalous. The common anomalies we consider include snow fault, open circuit, and partial occlusions due to leaves. Although, others factors such as partial shading also result in the loss of energy, we do not consider shade to be an anomaly, since it is a transient phenomena and does not need corrective action.

Our home dataset already includes real snow faults that are labelled and we evaluate the accuracy of our classifier on identifying these snow faults. We then use the synthetic faults from our solar anomaly dataset and synthetically inject them into the home dataset by introducing synthetic single panel faults as well as concurrent fault and evaluate the accuracy of our classifier. Figure 10 presents per-panel data for a typical day when electric fault or object covering anomaly has been injected into one or many panels.

5.3.1 Snow Fault Detection. We first evaluate the ability of the baseline method and our classifier in detecting snow faults in the home dataset. Recall that the dataset is labelled as normal or snow for each panel. We extract the features from daily power output, which include the

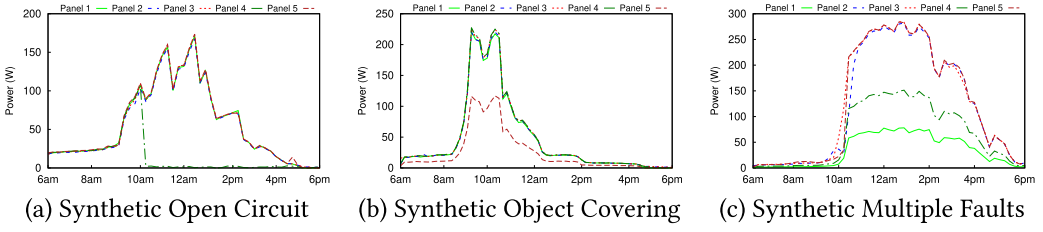


Fig. 10. Synthetic fault injection with (a) open circuit fault, (b) leaves covering fault, and (c) multiple leaves covering faults.

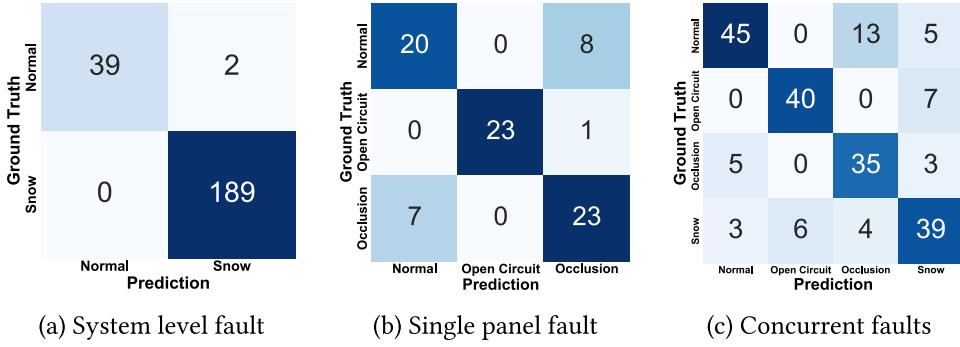


Fig. 11. Classification accuracy for our baseline linear regression anomaly detection method for (a) system-level faults, (b) single panel faults, and (c) multiple panel faults.

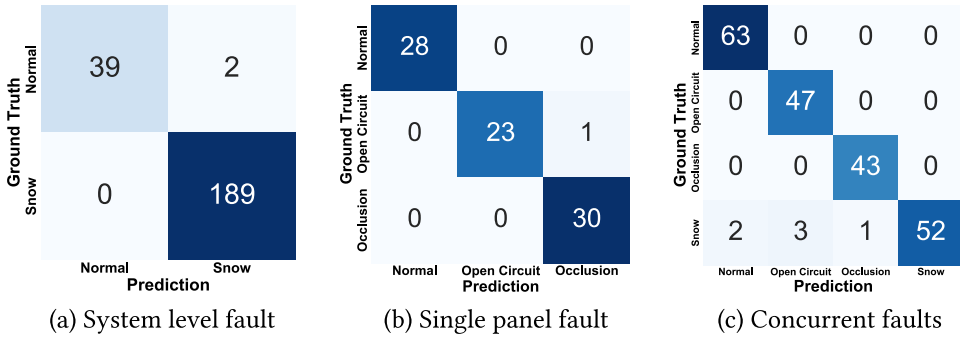


Fig. 12. Classification accuracy for (a) systemwide snow faults, (b) single panel faults, and (c) multiple panel faults.

Pearson's correlation coefficient, ratio of maximum observed power and the nominal panel capacity, and weather data, such as snow and cloud cover, and use them as inputs to our random forest classifier. Figure 11(a) and Figure 12(a) show the confusion matrix of the baseline method and our classifier, respectively. Both our approach and the baseline method have the same observed power output at the system-level and therefore have the same accuracy, since the two approaches only differ in the anomaly detection part of the approach. Table 2 and Table 3 shows that both methods are able identify system-level snow faults with an accuracy of 99.13%, precision of 0.9859, sensitivity of 1.00, and the F1 score of 0.9947. The high values for precision, recall, and F1 score means that even the baseline method does really well when identifying the system-level faults.

Table 2. Classification Metrics for Our Baseline Linear Regression Anomaly Detection Method

Classification	Accuracy	Precision	Recall	F1 score
Single, panel-level (open-circuit)	98.78%	1.00	0.9583	0.9787
Single, panel-level (occlusion)	80.49%	0.7188	0.7667	0.7419
Multiple, panel-level (open-circuit)	93.84%	0.8696	0.8511	0.8602
Multiple, panel-level (occlusion)	88.15%	0.6731	0.8140	0.7368
Multiple, panel-level (snow)	86.73%	0.7222	0.7500	0.7358
System level (snow)	99.13%	0.9895	1.00	0.9947

Table 3. Classification Metrics for Our Model-based Approach

Classification	Accuracy	Precision	Recall	F1 score
Single, panel-level (open-circuit)	98.78%	1.00	0.9583	0.9787
Single, panel-level (occlusion)	98.78%	0.9677	1.00	0.9836
Multiple, panel-level (open-circuit)	98.56%	0.9400	1.00	0.9691
Multiple, panel-level (occlusion)	99.53%	0.9773	1.00	0.9885
Multiple, panel-level (snow)	97.16%	1.00	0.8966	0.9455
System level (snow)	99.13%	0.9895	1.00	0.9947

We note that snow faults seen in our dataset tend to be systemwide faults, where all panels get covered with snow after a snow event and exhibit a snow fault concurrently. While it is certainly possible for only some panels to have snow cover (e.g., if snow melts unevenly across panels), our dataset presently does not have such faults.

5.3.2 Single and Concurrent Fault Classification. Since all observed snow faults in our dataset were system faults, we next show that our approach is still capable of fine-grain anomaly detection and classification of a single fault, and that it is also capable of detecting concurrent faults in a subset of the panels.

To do so, we use our solar anomaly dataset and choose the partial occlusion and open circuit anomaly from the dataset and inject these faults into a single, randomly chosen, panel of the array, where different panels have faults injected into them on different days. We first use the baseline method and our random forest classifier to identify the type of fault. We next use our model to detect the presence of the fault and our random forest classifier to identify the type of fault. We next inject multiple concurrent faults of all types—snow, occlusion, and open circuit—into the array using a similar methodology and attempt to detect and classify each fault using our model and classifier. Note that we need to use our concurrent fault detection approach in this case. For multi-panel fault classification using the baseline method, we use the per-panel baseline fault detection method for all the panels individually.

Figures 11(b) and 12(b) show the confusion matrix of classifying single faults in the array using the baseline method and our proposed approach. Table 2 and Table 3 show that both the baseline method and our approach show the same, high accuracy for single-panel open-circuit faults at 98.78%. The values for precision, recall, and F1 score are 1.00, 0.9583, and 0.9787, respectively. Both approaches yield the same result as open-circuit faults are very easy to identify. The real benefit of our approach is evident for single-panel occlusion faults where our model achieves a 98.78% accuracy as compared to 80.49% accuracy for the baseline method. Our approach also improves the precision from 0.7188 to 0.9677, recall from 0.7667 to 1.00, and F1 score from 0.7419 to 0.9836.

Figures 11(c) and 12(c) show the confusion matrix of classifying multi-panel faults in the array using the baseline method and our proposed approach. Table 2 and Table 3 show that both the

baseline method and our approach have a significantly higher accuracy for single-panel open-circuit faults at 98.56% for our approach and 93.84% for the baseline linear regression. However, as with single-panel faults, our approach shows the most significant improvement when classifying more complex faults, such as occlusion and snow. The accuracy for occlusion faults increases from 88.15% to 99.53%, while the accuracy for the snow faults increases from 86.73% to 97.16%. Our approach also improves the F1 score for the open-circuit, occlusion, and snow anomalies from 0.8602, 0.7368, and 0.7358 to 0.9691, 0.9885, and 0.9455, respectively. Table 2 and Table 3 show the values for all the classification metrics for both approaches. Sundown yields up to 30% improvement over the baseline regression method for some metrics and at least 8% improvement across all metrics the improvement is greater for more complex faults where baseline regression incurs higher errors. Also, the high values of precision, recall, and F1 score mean that our approach not only correctly classifies all the anomalous events to the correct category, but also does not misattribute non-anomalous events to any anomaly.

Results. Our experiments demonstrate the efficacy of our fault detection and classification methods for real snow faults as well as synthetically injected single and concurrent panel-level faults. Our results show that the random forest classifier is an effective approach for identifying both systemwide faults as well as faults that occur on a subset of panels. Our approach is able to classify snow, partial occlusion and open circuit faults with accuracy of more than 97% in terms of overall accuracy, specificity, and sensitivity.

6 RELATED WORK

There has been significant work on predicting power output for solar sites [7, 9, 15, 28, 29, 32, 34]. All of these studies predict only system-level output and generally report 20–30% error. These high errors and inability to predict panel-level output would cause their prediction for all panels to be the same, and limit their ability to detect and classify anomalies. There is also significant prior work on anomaly detection and classification in solar photovoltaic systems, which can be broadly classified into model-based approaches [14, 17, 20, 24, 25] and machine learning-based [10, 13, 16, 18, 26, 27, 30, 31, 36, 37] approaches. Some of these studies use power data from nearby solar sites [21, 35] to detect and classify anomalies. In Reference [35], authors compare the performance of different solar arrays at the same site, but do not do anomaly classification. Our work uses the output of other nearby panels to predict a panel's output for detecting faults and can classify various types of faults, i.e., snow, object covering, and electrical faults, on a single or multiple panels.

7 FUTURE WORK

We are working on extending our approach in multiple directions as a part of future work. Our evaluation in this article focuses on residential solar installations. However, our approach is also applicable to utility-scale systems as well. Utility-scale systems present both opportunities and challenges for SunDown's approach. More data are often available from utility-scale arrays, such as voltage and current, which could be leveraged by SunDown. Further, utility-scale deployments are often optimized with less shading and consistent geometries that are easier for SunDown to handle. At the same time, unlike residential-scale arrays that use micro-inverters to provide per-panel generation, utility-scale installations use string inverters and may lack per-panel output information. This will require our approach to be generalized to compare outputs on each "string" of panels, rather than each panel individually. This makes the approach more tractable to larger arrays with thousands of panels but also provides more coarse-grained anomaly information at the granularity of groups of panels. Nevertheless, we expect the approach to be useful for utility-scale systems that comprise thousands of panels, especially to decide what panels are in need of

cleaning or repair. For example, to minimize maintenance costs, an operator may not necessarily want to send out a technician until a certain number of panels are experiencing a fault.

Our anomaly detection approach requires a few days data to accurately predict per-panel solar output. However, the anomaly classifier used in our approach needs a significant number of anomalous events in each category for training purposes. This would require that the site has a long history available. Our future work involves the design of a “universal” classifier that can be trained using a collection of existing sites, and applied to new sites without additional training. We also plan to include other types of anomalies, such as dust and other types of electrical faults.

8 CONCLUSIONS

In this article, we proposed SunDown, a sensorless approach to detecting per-panel anomalies in residential solar arrays. Our approach uses a model-driven approach that leverages correlations between the power produced by adjacent panels to detect deviations from expected behavior. SunDown can handle faults in multiple panels and determine the probable cause of anomalies. We evaluated SunDown using two year panel-level generation data from the from a real site and a manually gathered dataset of various faults. Our approach requires data from only five panels for accurate prediction, is agnostic to weather characteristics, and yields high accuracy even when panels from different roof geometries are used. We show that our approach is accurate in predicting panel-level output with a MAPE of 2.98% and can correctly classify anomalies with >97% accuracy. We released the per-panel dataset from the real site and the manually generated dataset of various faults for research use.

REFERENCES

- [1] Solar Market Insight Report. 2018. Year In Review. Retrieved March 2020 from <https://www.seia.org/research-resources/solar-market-insight-report-2018-year-review>.
- [2] Solar Market Insight Report. 2019. Q4. Retrieved March 2020 from <https://www.seia.org/research-resources/solar-market-insight-report-2019-q4>.
- [3] Enphase Microinverters. Retrieved from <https://enphase.com/en-us/products-and-services/microinverters>.
- [4] SolarEdge Power Optimizer. Retrieved from <https://www.solaredge.com/products/power-optimizer>.
- [5] Y. Akiyama, Y. Kasai, M. Iwata, E. Takahashi, F. Sato, and M. Murakawa. 2015. Anomaly detection of solar power generation systems based on the normalization of the amount of generated electricity. In *Proceedings of the IEEE 29th International Conference on Advanced Information Networking and Applications* (2015).
- [6] M. Alsafasfeh, I. Abdel-Qader, B. Bazuin, Q. Alsafasfeh, and W. Su. 2018. Unsupervised fault detection and analysis for large photovoltaic systems using drones and machine vision. *Energies* 11, 9 (2018).
- [7] R. Andrews, J. Stein, C. Hansen, and D. Riley. 2014. Introduction to the open source PVlib for python photovoltaic system modelling package. In *Proceedings of the IEEE Photovoltaic Specialist Conference* (2014).
- [8] A. Arenella, A. Greco, A. Saggese, and M. Vento. 2017. Real time fault detection in photovoltaic cells by cameras on drones. In *Proceedings of the International Conference on Image Analysis and Recognition* (2017), Springer.
- [9] N. Bashir, D. Chen, D. Irwin, and P. Shenoy. 2019. Solar-TK: A. Data-driven toolkit for solar PV performance modeling and forecasting. In *Proceedings of the IEEE International Conference on Mobile Ad-hoc and Smart Systems (MASS'19)*.
- [10] M. Benedetti, F. Leonardi, F. Messina, C. Santoro, and A. Vasilakos. 2018. Anomaly detection and predictive maintenance for photovoltaic systems. *Neurocomputing* (2018).
- [11] D. Chen and D. Irwin. 2017. Black-box solar performance modeling: Comparing physical, machine learning, and hybrid approaches. *ACM SIGMETRICS Performance Evaluation Review* (2017).
- [12] D. Chen and D. Irwin. 2017. Sundance: Black-box behind-the-meter solar disaggregation. In *Proceedings of the ACM International Conference on Future Energy Systems (e-Energy'17)*.
- [13] W. Chine, A. Mellit, V. Lughi, A. Malek, G. Sulligoi, and A. A. Pavan. 2016. Novel fault diagnosis technique for photovoltaic systems based on *Artificial Neural Networks*. *Renew. Energy* (2016).
- [14] M. Dhimish, V. Holmes, and M. Dales. 2017. Parallel fault detection algorithm for grid-connected photovoltaic plants. *Renew. Energy* (2017).
- [15] N. Engerer and F. Mills. 2014. Kpv: A. Clear-sky index for photovoltaics. *Solar Energy* 105 (July 2014).

- [16] P. Gao, L. Golab, and S. Keshav. 2015. What's wrong with my solar panels: A data-driven approach. In *Proceedings of the International Conference on Extending Database Technology/International Conference on Database Theory (EDBT/ICDT'15) Workshops*.
- [17] E. Garoudja, F. Harrou, Y. Sun, K. Kara, A. Chouder, and S. Silvestre. 2017. Statistical fault detection in photovoltaic systems. *Solar Energy* (2017).
- [18] F. Harrou, A. Dairi, B. Taghezouit, and Y. Sun. 2019. An unsupervised monitoring procedure for detecting anomalies in photovoltaic systems using a one-class support vector machine. *Solar Energy* (2019).
- [19] B. Hu. 2012. *Solar Panel Anomaly Detection and Classification*. MS Thesis, University of Waterloo, Canada.
- [20] Y. Hu, B. Gao, X. Song, G. Tian, K. Li, and X. He. 2013. Photovoltaic fault detection using a parameter based model. *Solar Energy* (2013).
- [21] S. Iyengar, S. Lee, D. Sheldon, and P. Shenoy. [n. d.]. Solarclique: Detecting anomalies in residential solar arrays. In *Proceedings of the ACM SIGCAS Conference on Computing and Sustainable Societies*. pp. 1–10.
- [22] S. Iyengar, N. Sharma, D. Irwin, P. Shenoy, and K. Ramamritham. 2017. A cloud-based black-box solar predictor for smart homes. *ACM Trans. Cyber-phys. Syst.* (2017).
- [23] S. Kalayaraman. 2016. *Cognitive Drones: Aerial Robotics and Nano Satellites in Cognitive iot Solutions*. IBM Research Report.
- [24] B. Kang, S. Kim, S. Bae, and J. Park. 2012. Diagnosis of output power lowering in a PV array by using the kalman-filter algorithm. *IEEE Trans. Energy Convers* (2012).
- [25] K. Kim, G. Seo, B. Cho, and P. Krein. 2015. Photovoltaic hot-spot detection for solar panel substrings using AC parameter characterization. *IEEE Trans. Power Electr.* (2015).
- [26] G. Liu and W. A. Yu. 2017. Fault detection and diagnosis technique for solar system based on elman neural network. In *Proceedings of the IEEE Information Technology, Networking, Electronic and Automation Control Conference (ITNEC'17)*. IEEE.
- [27] G. Liu, W. Yu, and L. Zhu. 2018. Condition classification and performance of mismatched photovoltaic arrays via a pre-filtered elman neural network decision making tool. *Solar Energy* (2018).
- [28] E. Lorenz, D. Heinemann, H. Wickramaratne, H. Beyer, and S. Bofinger. 2007. Forecast of ensemble power production by grid-connected pv systems. In *Proceedings of the 20th European PV Conference* (2007), Milano.
- [29] E. Lorenz, J. Hurka, D. Heinemann, and H. Beyer. 2009. Irradiance forecasting for the power prediction of grid-connected photovoltaic ssystems. *IEEE J. Select. Top. Appl. Earth Observ. Remote Sens.* (2009).
- [30] H. Mekki, A. Mellit, and H. Salhi. 2016. Artificial neural network-based modelling and fault detection of partial shaded photovoltaic modules. In *Simulation Modelling Practice and Theory* (2016).
- [31] J. Pereira and M. Silveira. 2018. Unsupervised anomaly detection in energy time series data using variational recurrent autoencoders with attention. In *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA'18)*, IEEE.
- [32] R. Perez, J. Schlemmer, S. Kivalov, J. Dise, P. Keelin, M. Grammatico, T. Hoff, and A. A. Tuohy. 2018. New version of the suny solar forecast model: A scalable approach to site-specific model training. In *Proceedings of the IEEE 45th Photovoltaic Specialists Conference (PVSC'18)*.
- [33] B. Schölkopf, D. Hogg, D. Wang, D. Foreman-Mackey, D. Janzing, C. Simon-Gabriel, and J. Peters. 2016. Modeling confounding by half-sibling regression. *Proc. Natl. Acad. Sci. U.S.A.* (2016).
- [34] N. Sharma, P. Sharma, D. Irwin, and P. Shenoy. 2011. Predicting solar generation from weather forecasts using machine learning. In *Proceedings of the IEEE International Conference on Smart Grid Communications (SmartGridComm'11)*.
- [35] S. Vergura. 2018. Hypothesis tests-based analysis for anomaly detection in photovoltaic systems in the absence of environmental parameters. *Energies* (2018).
- [36] Y. Zhao, Q. Liu, D. Li, D. Kang, Q. Lv, and L. Shang. 2018. Hierarchical anomaly detection and multimodal classification in large-scale photovoltaic systems. *IEEE Trans. Sust. Energy* (2018).
- [37] H. Zhu, L. Lu, J. Yao, S. Dai, and Y. Hu. 2018. Fault diagnosis approach for photovoltaic arrays based on unsupervised sample clustering and probabilistic neural network model. *Solar Energy* (2018).

Received August 2020; revised March 2021; accepted April 2021