Probabilistic Analysis of Solar Cell Performance Using Gaussian Processes

Rahul Jaiswal, Manel Martínez-Ramón, and Tito Busani

***Abstract*—This article investigates the application of machine learning-based probabilistic prediction methodologies to estimate the performance of silicon-based solar cells. The concept of confidence-bound regions is introduced and the advantages of this concept are discussed in detail. The results show that the optical and electrical performance of a photovoltaic device can be accurately estimated using Gaussian processes with accurate knowledge of the uncertainty in the prediction values. It is also shown that cell design parameters can be estimated for a desired performance metric and trained machine learning models can be deployed as a standalone application.**

***Index Terms*—Gaussian processes, machine learning, PERC cell, photovoltaics, TCAD simulation.**

I. Introduction

MACHINE learning (ML)-based methods are being used to optimize photovoltaic device design and fabrication recipes [1], [2], which has shown to be better in terms of resources, manpower, and time expenditure than traditional loss optimization strategies, which involve repetitions of experiments to yield quantitatively varying datasets. However, there are factors [3] that restrict the acceptance of using ML model predictions for device and material optimization, like their accuracy, which is generally estimated after model training on a test database, and generalization [4], which is unknown until compared with characterization/simulation data.

Efficiency of ML model implementation in photovoltaics research has been improved in prior works using techniques like transfer learning [5]; in this research, Gaussian processes for ML [6] are used. In particular, we apply Gaussian process regression (GPR). The GPR methodology is based on a Gaussian model for the training data error, which leads to a Gaussian likelihood function for the regressors and, together with a Gaussian assumption for the prior distribution of the model parameters, it leads to a model for the prediction that includes a mean and a variance. This is advantageous over other regression approaches because, provided the likelihood and the prior assumptions are correct, they give not only a prediction but also a confidence interval over this prediction that allows to determine whether its quality is acceptable or not for the application at hand [7]. Also, the nature of GPR is such that it does not have free parameters, so no cross-validation is needed in the training process.

Using GPRs ensures some degree of confidence that any conclusions made during predictions are robust to the extent of uncertainty in the data, i.e., it makes sure that the inferred parameter (cell performance parameter like reflection profile or cell design parameter like texture angle) is not specific to the particular noisy dataset that was used for training the model, and a confidence bound for a given number of standard deviations can be drawn over the regression predictions. These confidence bounds for each prediction can be monitored to dynamically inform decisions about when to trust a trained model predictions in a high-throughput environment. In our work, GPR prediction mean values are also compared against the prediction of other regression models like ensemble techniques and neural networks. As other ML models do not provide a prediction of confidence bound region (uncertainty values), the comparison is made only between the mean values of the Gaussian process regression model in the graphs shown in the results section. Gaussian process regression is an extension of the kernel regression, so to predict just the mean prediction, either of the strategies can be used.

Complete numerical simulation analysis for a solar cell can be divided into optical and electrical simulation, as the unit cell for a textured cell in electrical domain will contain a large number of repeating pyramid structures. In this article, ML model equivalents for optical simulations are developed and uncertainty of model prediction is studied for different test cases (different cell design and material parameters). Model generalization for other training datasets (experimental data) is discussed. Apart from forward prediction (i.e., outcome of a process step, given the process parameters and the results of the previous process step), the efficiency of back-prediction (predicting the change in design/material parameters to achieve a desired process result) is also explored. In the related literature, methods like Bayesian inference [8] and autoencoder neural network architectures [9] have been used to identify process parameters. In our work, we are predicting process parameters, along with their prediction

Manuscript received September 21, 2021; revised October 29, 2021; accepted January 6, 2022. Date of publication February 1, 2022; date of current version February 19, 2022. The work Manel Martínez-Ramón was supported in part by NSF EPSCoR under Grant OIA-1757207 and in part by the King Felipe VI endowed Chair. *(Corresponding author: Tito Busani.)*

Rahul Jaiswal and Tito Busani are with the University of New Mexico Center for High Technology Materials, Albuquerque, NM 87106 USA, and also with the Electrical and Computer Engineering Department, University of New Mexico, Albuquerque, NM 87131 USA (e-mail: rahul17455@gmail.com; busanit@unm.edu).

Manel Martínez-Ramón is with the Electrical and Computer Engineering Department, University of New Mexico, Albuquerque, NM 87131 USA (e-mail: Martinez-Ramonmanel@unm.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JPHOTOV.2022.3143457.

Digital Object Identifier 10.1109/JPHOTOV.2022.3143457

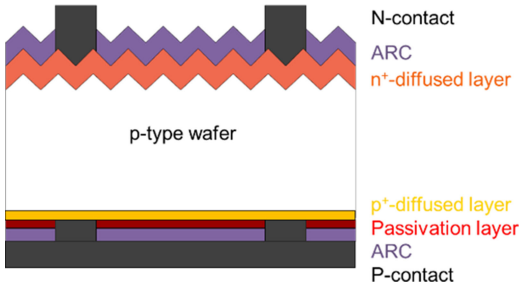

Fig. 1. PERC cell.

uncertainty. Finally, a strategy to deploy trained ML models is discussed.

II. Methodology

For calculating the optical performance of solar cells, the software Sentaurus device TCAD is used. Python was used for ML model development and statistical analysis. This work involves three main steps. Initially, a set of variables (cell design and material parameters) for optical simulation are used as inputs for the Sentaurus model to simulate the solar cell optical performance (characterized by optical generation in the cell and reflectance from the cell). Using this dataset (Simulation parameters and corresponding outputs), ML models are trained. Finally, these models are used to predict cell performance for certain test cases, and besides the prediction accuracy, the uncertainty of prediction is observed for test cases that are within the training dataset range and outside of it. Confidence interval in these predictions are also quantified for input test points that are close to training points and for those which are comparatively far away.

A. Device Model

A p-PERC cell with totally diffused rear [10] (Fig. 1) was the target design that was chosen in this study. A 3-D Sentaurus TCAD model was designed for optical simulation on a single-side textured structure (with transfer matrix method applied for the thin front nitride layer and ray tracing for rest of the wafer) and free carrier absorption [11] was enabled. Among all the parameters that effect the optical structure, we choose the following six parameters, with known physical effects on the cell optical performance to be varied in the simulation for creating a simulation database:

1) Wafer Thickness: We are simulating two performance metrics: reflection profile and depth-dependent carrier generation, which accommodates the effects of incident light rays of shorter and longer wavelength absorption for different cell thicknesses [13]. Both the optical generation and the reflection from the cell are dependent on the path length that the incident light travels within the wafer.

2) Substrate Doping: Substrate doping is varied as an input to the ML model. However, the emitter window (doped) layer is defined by a Gaussian profile with peak value of 2×10^{20} cm^{-3}. Free carrier absorption within the wafer has a dependency on carrier concentration [12] and it is observed only for high doping concentrations (generally, doped regions for a solar cell). We have included this parameter in our study for two reasons, first to see if the ML predictions agree with the knowledge from the related literature and simulation results and, second, the result of this optical simulation prediction/simulation will be fed into an electrical simulation/prediction model; this way, the compensating effects of a parameter change in two prediction models can be studied in future.

3) Front Textured Pyramid Angle: The simulation unit cell was composed of upright regular pyramids. This unit cell had perfect reflecting boundaries (to virtualize the full cell structure). Pyramid structures help in incident light trapping via multiple reflections and total internal reflection [14]. The incident light scattering will be different for different pyramid dimensions, depending on the pyramid base angle.

4) Rear-Side Contact (Aluminum) Thickness: The aluminum–silicon interface at the rear of Si solar cells absorbs long-wavelength photons not collected in the bulk (silicon absorber), eliminating the transmittivity through the cell. Variations in the thickness of this contact should not have any effect on cell optical performance. This concept will be studied by comparing the ML predictions to simulation results during testing. Long-wavelength photons may be converted into heat and reduce the operating device efficiency. This concept can be tested in future work using a multiphysics simulation model.

5) Front-Side Antireflective Coating Thickness: The amount of reflectivity from an antireflective material-coated surface is dependent on thickness of the anti-reflection coating (ARC), which, in turn, is dependent on the wavelength of the incoming wave and refractive indices of the materials involved. Variations in this thickness can help to identify the optimized ARC thickness [15] for a particular cell design rapidly using ML models for a polychromatic light source.

6) Back-Reflectivity: The rear-side dielectric layer in PERC cells does not just passivate the surface and improves electrical performance, but it reflects the photons that have not been involved in electron hole generation, essentially giving them a second chance to increase the optical generation in the cell. It also contributes to further randomization of light (within the cell). After identifying the optimal dielectric layer material and thickness for rear side, its effect (lumped by the back-reflectivity parameter) on improving the cell optical performance [16] can be prototyped rapidly using a ML model.

B. Data Preparation

These six input parameters of the simulation models were statistically varied to create 768 simulations. We varied wafer thickness in five steps between 150 and 300 μm. Substrate doping was varied between 1.5×10^{15} cm^{-3} and 10^{16} cm^{-3} in three steps, textured pyramid angles were varied from 32 degrees to 33.5 degrees in four steps, aluminum thickness was varied between 20 and 40 μm, ARC thickness was varied between 60 and 70 nm, and the back reflectivity was varied between 60% and 90%. The simulation output for the depth dependent carrier generation profile is a 2-D list, where the first column contains depth points within the wafer, and the second column

contains the carrier generation at the corresponding depth points. The value of the last row in the first column (depth points) is equal to the wafer thickness (i.e., one of the input parameters for the simulation). One simulation corresponds to an input matrix of rank $[m \times n]$, where $m = 6$ (i.e., the number of simulation inputs) and $n = 1$ (one simulation), while the output matrix has a rank of $[r \times s]$, where $r = 2$ (i.e., two columns, depth, and carrier generation) and s is a number distributed between 0 and the wafer thickness value.

The ML regression models predict one parameter, so we need to flatten the output list of the simulations to create a training database. The depth points from the simulation output were used as one of the inputs in the training database (apart from the existing six input parameters). Therefore, one simulation of input rank $[m \times n]$ and output rank $[r \times s]$ correlates to an input matrix of dimensions $[(m + 1) \times s]$ and an output matrix of $[1 \times s]$ in the training database. The input parameters in one simulation are padded s times and a new column (depth points) is added to create the training database input, while the output matrix in the training database is just a 1-D list of carrier generation.

The simulation output for the reflection profile is also a 2-D list, where the first column contains the wavelength of the light incident on the wafer, and the second column contains the percentage of light rays reflected corresponding to that wavelength. One simulation corresponds to a input matrix of dimensions $[a \times b]$, where $a = 6$ (i.e., the number of simulation inputs) and $b = 1$ (one simulation), while the output matrix has dimensions $[o \times p]$, where $o = 2$ (i.e., two columns, wavelength, and reflectance) and $p = 18$ (the wavelength values are varied from 300 nm to 1.2 μm in steps of 50 nm). The output list of the simulations was flattened to create a training database. The wavelength points from the simulation output were used as one of the inputs in the training database. Therefore, one simulation of input dimensions $[a \times b]$ and output dimensions $[o \times p]$ correlates to an input matrix of dimensions $[(a + 1) \times p]$ and output matrix of dimensions $[1 \times p]$ in the training database.

The ML model accuracy is determined using the conventional test–train split, where we used 20% of full training dataset as test data to check the coefficient of determination or the r-squared (denoted by $R2$ in our work) score [17] between the prediction and the actual (simulation output) data. The r-squared score informs about what percent of the prediction error in the dependent variable is eliminated when a regression is performed on the independent variable. A third type of dataset called validation (or verification) is created (by performing additional simulations) to verify the model generalization, as the model occasionally sees this verification data, but it never learns from this. In other words, the ML model hyperparameters (and priors in the case of Gaussian processes) are tuned during the training to increase its prediction accuracy in the test data range. Therefore, it can develop a bias for it. Verification input sets are not part of either the training or test datasets and their values can be outside its range.

A total of 13 824 training data points for reflectance profile prediction and 172 000 data points for optical generation profile prediction were created. These data points were sampled during model training and testing.

C. Gaussian Process Regression Models

For reflectance profile prediction, the ML model will be trained on the training dataset created as described in the previous section. The reflectance at a given wavelength can be predicted at a time. By repeating the prediction for different wavelengths (keeping the other six parameters same), a complete reflection profile can be predicted.

Similarly, for optical generation profile prediction, generation at a given depth point is predicted at a time, and then, this process is repeated for different depth points (i.e., from the surface or "0" to substrate thickness).

In order to proceed with the prediction tasks, a GP regression model is used. The GP model is an estimator of the form [18]

$$y_i = \mathbf{w}^\top \phi(\mathbf{x}_i) + e_i \tag{1}$$

with $1 \leq i \leq N$, where y_i is the target to be predicted or regressor, $\mathbf{x}_i$ is the input observation or predictor, and e_i is the prediction error. From a GP standpoint, the error is considered a sequence of independent and identically distributed Gaussian samples of zero mean and variance σ_n^2. Function $\phi(\cdot)$ maps the input features into a higher dimensional Hilbert space endowed with a dot product $K(\mathbf{x}_i, \mathbf{x}_j) = \phi^\top(\mathbf{x}_i)\phi(\mathbf{x}_j)$. Function $K(\cdot, \cdot)$ is called a kernel, and, by virtue of the Mercer's theorem [19], the only condition for it to be a dot product in a higher dimension Hilbert space is that the function is definite positive. The Representer theorem [20] assures that there exists an equivalent representation of the model into a dual space expressed only as a linear combination of dot products, with the form

$$y_i = \sum_{j=1}^{N} \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) + e_i \tag{2}$$

where $\mathbf{x}_j$ is a set of training data, and α_j is a set of dual trainable parameters. A quantity of positive definite functions can be used as kernel so the estimator has nonlinear capabilities.

For our work, we have used the squared exponential kernel [21] (radial basis function) given by

$$K(\mathbf{x_i}, \mathbf{x_j}) = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x_i} - \mathbf{x_j})^\top \Theta^{-2}(\mathbf{x_i} - \mathbf{x_j})\right) \tag{3}$$

and the RQ kernel [22] given by

$$K(\mathbf{x_i}, \mathbf{x_i}) = \sigma_f^2 \left(1 + \frac{1}{2\alpha}(\mathbf{x_i} - \mathbf{x_j})^\top \Theta^{-2}(\mathbf{x_j} - \mathbf{x_j})\right)^{-\alpha} \tag{4}$$

where σ_f, α, and Θ (length-scale) are hyperparameters of the kernel functions.

The GP is solved by first stating a prior probability distribution $p(\mathbf{w})$ for the primal parameters $\mathbf{w}$, which is a multivariate standard, and a Gaussian conditional likelihood $p(y_j|\mathbf{x}_j, \mathbf{w})$ for the training data, with variance σ_n^2. By using the Bayes rule, a posterior distribution of the primal parameters is found. Then, a posterior distribution can be found for a test sample $\mathbf{x}^*$, which

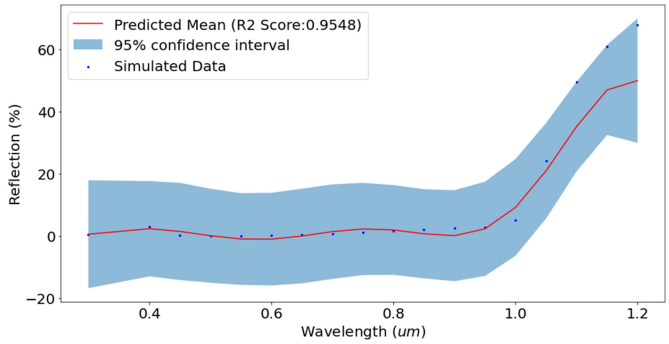

Fig. 2. Reflection profile prediction (GP model trained with three features).

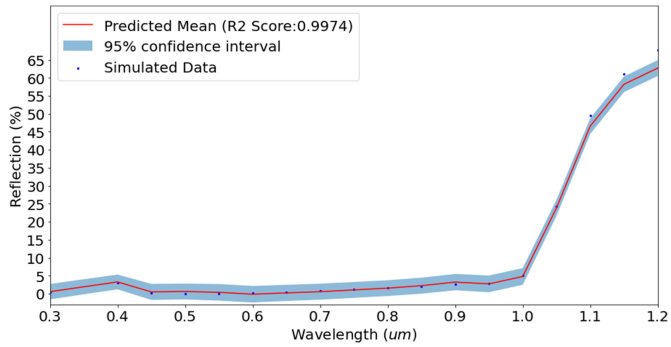

Fig. 3. Reflection profile prediction (GP model trained with all features).

is another Gaussian with mean and variance given by

$$\bar{f}(\mathbf{x}^*) = \mathbf{y}^\top \left(\mathbf{K} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{k}(\mathbf{x}^*) \tag{5}$$

and

$$\sigma_*^2 = k(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{k}^\top(\mathbf{x}^*) \left(\mathbf{K} + \sigma_n^2 \mathbf{I}\right)^{-1} \mathbf{k}(\mathbf{x}^*) \tag{6}$$

where $\mathbf{y}$ is a column vector containing all the training regressors, $\mathbf{K}$ is the kernel matrix of dot products between training predictors $K(\mathbf{x}_i, \mathbf{x}_j)$, $\mathbf{k}(\mathbf{x}^*)$ is a column vector containing the dot products $K(\mathbf{x}^*, \mathbf{x}_j)$ between the training data and the test sample, and $\mathbf{I}$ is an identity matrix. The variance gives a confidence interval over the prediction. The hyperparameters of the kernel and the noise parameter σ_n^2 are optimized by maximizing the marginal log likelihood of the training regressors with respect to them, which is usually done by gradient descent.

III. Results

Squared exponential or radial basis function (RBF) kernel was used for the Gaussian process model to predict the reflection profile. With only three features used for training the Gaussian process regression model for predicting the reflection profile, the accuracy of mean prediction and confidence bound will be affected. "Substrate thickness," "substrate doping," and "rear-side contact (aluminum) thickness" were the three parameters which were used to train the first model, and its prediction for a test case is shown in Fig. 2.

All six feature values were used for training the prediction accuracy for mean values increased from 0.9596 to 0.9974 and the confidence bound was narrowed as shown in Fig. 3.

Another expected result was obtained when the feature parameter "rear-side contact (aluminum) thickness" was excluded from the model training (and the rest of five features were used). It has no effect on the model prediction accuracy and confidence interval, which agrees with cell device physics, as variations in this parameter should not have any effect on the reflection profile of the wafer.

This same phenomenon is observed from ML models for optical generation profile prediction. For this model, a rational quadratic (RQ) kernel was used to calculate the covariance. With just three input features, the prediction accuracy was smaller than that observed for a model trained with all six features. The confidence interval was also bigger for the model with three input features. These two predictions for a test case are shown in Fig. 4. Similar to the prediction of reflection profile exclusion of the "rear-side contact (aluminum) thickness," feature value has no effect on the prediction accuracy of optical generation profile.

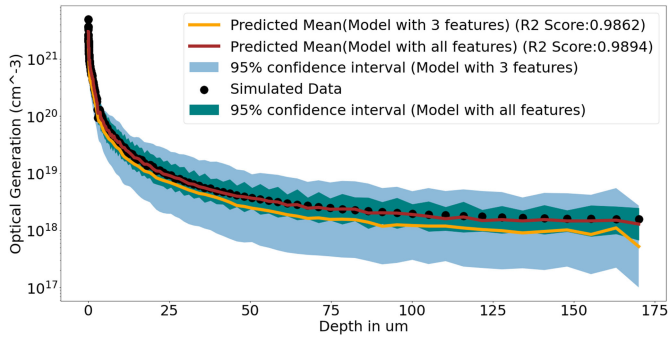

Fig. 4. Optical generation profile prediction comparison (GP model trained with all features vs. model trained with three features).

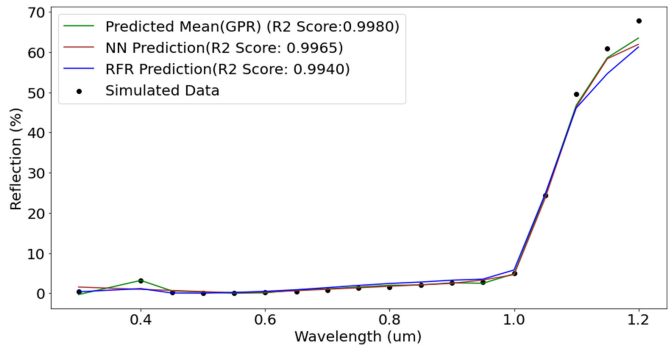

Fig. 5. Gaussian process, neural network, and random forest regression prediction for reflectance profile.

The GP prediction mean values were also compared with the predictions of other ML models. We designed an ensemble model using random forest regression [23] and a neural network regression model [24] (three layers; first two layers have rectified linear activations and a linear last layer). This comparison for reflectance profile prediction is shown in Fig. 5.

The comparison of Gaussian process prediction mean values for optical generation profile with neural network and random forest regression prediction is shown in Fig. 6.

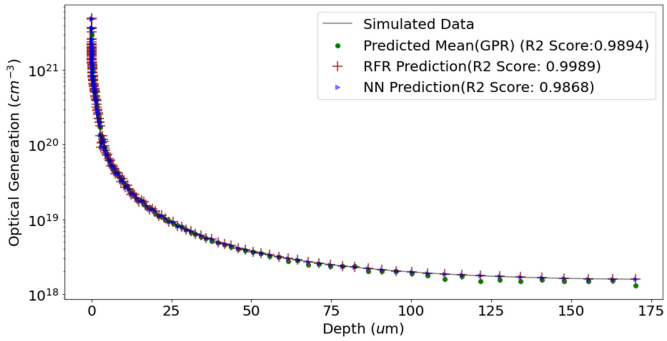

Fig. 6. Gaussian process, neural network, and random forest regression prediction for optical generation profile.

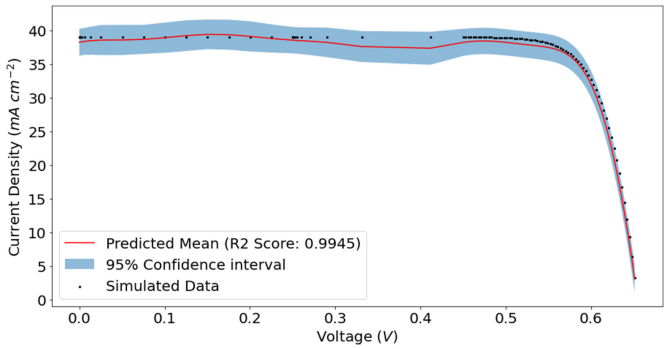

Fig. 7. Prediction (mean value and confidence interval) of cell current–voltage profile.

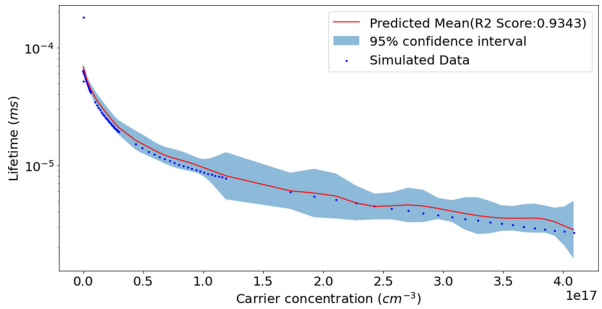

Fig. 8. Prediction (mean value and confidence interval) of minority carrier lifetime in the cell.

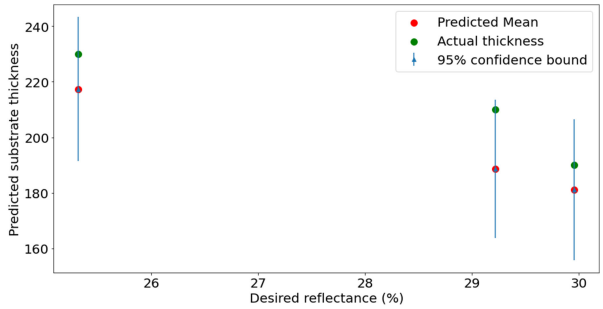

Fig. 9. Back-predicting wafer thickness necessary to achieve a certain reflection (at a given wavelength and other cell design constraint).

Among the compared methodologies (GP, random forests, and a neural network), only the GP strategy gives a probabilistic (Gaussian) distribution of the prediction that allows the user to compute a confidence interval, and this is the main reason for the choice. Indeed, the confidence interval depends on the hyperparameters of the model. Nevertheless, the main advantage of the GP method is that these parameters do not need to be cross-validated. Instead, these parameters are adjusted through the maximization of the training data likelihood distribution with respect to these hyperparameters. An important value to be optimized through ML is precisely the standard deviation of the estimation error. This and the kernel parameters appear in the predictive posterior variance estimation. This is helpful during fabrication of the device as the degree of risk (time, manpower, and resource investment) involved in every experiment (process parameters change) can be evaluated and compared in advance. GP regression provides the measure of uncertainty which is evaluated for every prediction. In contrast, the prediction mean accuracy for other ML model can only be evaluated during model training and validation. A case where need for measure of uncertainty becomes critical is when the input feature values of the ML model are not within the training data range. Then, the prediction mean values cannot always be trusted.

To test the performance of the GPR for predicting other cell performance parameters, we designed 2-D electrical device simulation models in Sentaurus TCAD to calculate the current–voltage and minority carrier lifetime profiles. In these device models, Fermi-dirac statistics were used for electron and holes, and the drift-diffusion model was used for carrier transport. Band gap narrowing was enabled for the substrate. For auger recombination, the Richter model [25] was used and the Philips unified mobility model [26] was used for carrier mobility calculations. The following parameters were statistically varied in these simulation models to create training databases: substrate doping, substrate thickness, carrier recombination velocities (in the front and rear surfaces), electron and hole lifetime in the wafer, peak rear diffusion, and front texture quality (pyramid angle).

GPR predictions of cell current–voltage profile and minority carrier lifetime profile for a test case are shown in Figs. 7 and 8, and these predictions are compared with their corresponding actual (simulated) values.

GP regression models can also perform back-prediction (to estimate cell design parameter for a desired performance metric). They can predict both the mean value and confidence interval (uncertainty in back prediction). We trained a model with six input parameters—pyramid angle, front-side ARC thickness, back-reflectivity, rear-side contact thickness, incident light wavelength, and the reflection from the cell at that wavelength. The output of the model was cell thickness. Basically, for a given wavelength, we can estimate the wafer thickness necessary to achieve a certain reflection (given other five parameters are kept constant for a single prediction). Fig. 9 demonstrates three such predictions for three different reflection values.

Once trained, the state (hyperparameters in case of Gaussian process regression) of the ML models can be saved as it is in a static external file (for example, a HDF5 file format). This model can then be wrapped as a REST application programming

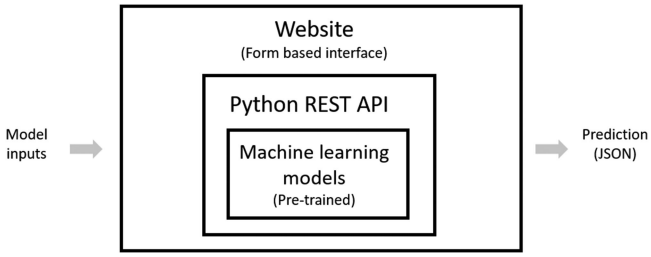

Fig. 10. Proposed framework for deployment of trained machine learning model.

interface (API) using frameworks like Python Flask, and can be made accessible via a web-interface [27] (for example, a website with form-based inputs, where users can enter the model features/cell design parameters) and the model will predict outputs on the fly by loading the precalculated model state. The outputs (prediction results) can be displayed either as raw data, for example, a JSON (Java Script Object notifier) object or a plot (image). The framework is described in Fig. 10. This deployment strategy has two main advantages. First, the computation is done on a network server (no software setup on the user-side) and, second, the model source code and training data are not exposed. We have designed one such API, currently running on our internal website.

IV. Discussion

ML model frameworks proposed in the literature [28] generally provide a conditional mean value as their prediction. In addition to the mean prediction value, GP also provides a confidence interval for cell performance parameters (optical generation, reflection, minority carrier lifetime, and current voltage profile).

The results obtained with different input features are in agreement with those features reported in previous literature to significantly affect the prediction quality. These features produce a better mean prediction and narrower confidence intervals. Excluding features that do not have effects in cell performance did not decrease the prediction quality.

Our analysis shows that GP is useful in cell electrical performance parameter prediction. It is important to see that when the input features with severe sparsity, test predictions with features in the training range will produce large confidence intervals. This is observed in current–voltage profile prediction. The training data has less data points between 0.2 and 0.45 V, and the confidence interval is wider. The mean predicted values are also less accurate, in agreement with the estimated predictive confidence interval. This phenomenon is even more exaggerated in the model for predicting minority carrier lifetime, as there is no training data points around carrier concentration of 1.5×10^{17} $cm^{-}3$, so the prediction confidence interval is very wide around this point.

V. Conclusion

GP predicts solar cell electrical and optical parameters over a wide range of input parameters and the prediction quality depends on training data and input features. These models can be retrained using experimental (measured data) or a combination of simulation and characterization data for a given cell architecture. Given a performance metric value, it is also possible to back-predict cell design parameters with uncertainty in the prediction. These GP model features constitute a tool in an academic or industrial setup where variation in performance of a particular cell architecture can be predicted with a confidence bound by changing its design parameters.

We are designing 3-D electrical simulation models in Sentaurus TCAD and their digital twins. In the future, we plan to present a pipelined structure where optical performance predictions from one ML model will be fed into other ML models predicting electrical performance, thereby investigating the compensation of the effect of cell design parameters in the electrical and optical domains.

References

[1] H. Wagner-Mohnsen and P. Altermatt, "A combined numerical modeling and machine learning approach for optimization of mass-produced industrial solar cells," *IEEE J. Photovolt.* vol. 10, no. 5, pp. 1441–1447, Sep. 2020, doi: 10.1109/jphotov.2020.3004930.

[2] M. Kaya and S. Hajimirza, "Rapid optimization of external quantum efficiency of thin film solar cells using surrogate modeling of absorptivity," *Sci. Rep.*, vol. 8, pp. 1–9, 2018, doi: 10.1038/s41598-018-26469-3.

[3] Y. Liu, T. Zhao, W. Ju, and S. Shi, "Materials discovery and design using machine learning," *J. Materiomics* vol. 3, no. 3, pp. 159–177, 2017, doi: 10.1016/j.jmat.2017.08.002.

[4] Y. Chung, P. J. Haas, E. Upfal, and T. Kraska, "Unknown examples & machine learning model generalization," Oct. 2019, *arXiv:1808.08294 [cs]*.

[5] M. Kaya and S. Hajimirza, "Using a novel transfer learning method for designing thin film solar cells with enhanced quantum efficiencies," *Sci. Rep.*, vol. 9, pp. 1–10, 2019, doi: 10.1038/s41598-019-41316-9.

[6] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2008.

[7] S. Seo, M. Wallat, T. Graepel, and K. Obermayer, "Gaussian process regression: Active data selection and test point rejection," in *Proc. IEEE-INNS-ENNS Int. Joint Conf. Neural Networks. IJCNN 2000. Neural Comput.: New Challenges Perspectives New Millennium*, 2000, pp. 241–246, doi: 10.1109/IJCNN.2000.861310.

[8] Z. Ren *et al.*, "Embedding physics domain knowledge into a bayesian network enables layer-by-layer process innovation for photovoltaics," *NPJ Comput. Mater.*, vol. 6, no. 1, pp. 1–9, 2020, doi: 10.1038/s41524-020-0277-x.

[9] Z. Ren *et al.*, "Physics-guided characterization and optimization of solar cells using surrogate machine learning model," in *Proc. IEEE 46th Photovolt. Specialists Conf.*, 2019, pp. 3054–3058, doi: 10.1109/PVSC40753.2019.8980715.

[10] S. Meier, S. Wasmer, A. Fell, N. Wöhrle, J. Greulich, and A. Wolf, "Efficiency potential of p-type pert vs. perc solar cells," in *Proc. IEEE 7th World Conf. Photovolt. Energy Convers.*, 2018, pp. 3578–3583, doi: 10.1109/PVSC.2018.8547393.

[11] D. K. Schroder, R. N. Thomas, and J. C. Swartz, "Free carrier absorption in silicon," *IEEE J. Solid-State Circuits*, vol. 13, no. 1, pp. 180–187, Feb. 1978, doi: 10.1109/JSSC.1978.1051012.

[12] D. A. Clugston and P. A. Basore, "Modelling free-carrier absorption in solar cells," *Prog. Photovolt.: Res. Appl.* vol. 5, no. 4, pp. 229–236, 1997, doi: 10.1002/(SICI)1099-159.

[13] A. Zaki and A. El-Amin, "Effect of cell thickness on the electrical and optical properties of thin film silicon solar cell," *Opt. Laser Technol.* vol. 97, pp. 71–76, 2017, doi: 10.1016/j.optlastec.2017.06.009.

[14] V. Moroz, J. Huang, K. Wijekoon, and D. Tanner, "Experimental and theoretical analysis of the optical behavior of textured silicon wafers," in *Proc. 37th IEEE Photovolt. Specialists Conf.*, 2011, pp. 2900–2905, doi: 10.1109/PVSC.2011.6186552.

[15] S. Duttagupta, F. Ma, B. Hoex, T. Mueller, and A. G. Aberle, "Optimised antireflection coatings using silicon nitride on textured silicon surfaces based on measurements and multidimensional modelling," *Energy Procedia*, vol. 15, pp. 78–83, 2012, doi: 10.1016/j.egypro.2012.02.009.

[16] Z. C. Holman *et al.*, "Parasitic absorption in the rear reflector of a silicon solar cell: Simulation and measurement of the sub-bandgap reflectance for common dielectric/metal reflectors," *Sol. Energy Mater. Sol. Cells*, vol. 120, pp. 426–430, 2014, doi: 10.1016/j.solmat.2013.06.024.
[17] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination r-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Comput. Sci.* vol. 7, pp. e623–e647, 2021, doi: 10.7717/peerj-cs.623.
[18] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School Machine Learning*. Berlin, Germany: Springer, 2003, pp. 63–71.
[19] J. Mercer, "XVI functions of positive and negative type, and their connection the theory of integral equations," *Philos. Trans. Roy. Soc. London. Ser. A.*, vol. 209, nos. 441–458, pp. 415–446, 1909.
[20] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," in *Proc. Int. Conf. Comput. Learn. Theory*, 2001, pp. 416–426.
[21] N. Ulapane, K. Thiyagarajan, and S. Kodagoda, "Hyper-parameter initialization for squared exponential kernel-based Gaussian process regression," in *Proc. 15th IEEE Conf. Ind. Electron. Appl.*, 2020, pp. 1154–1159, doi: 10.1109/ICIEA48937.2020.9248120.
[22] A. Solin and S. Särkkä, "Gaussian quadratures for state space approximation of scale mixtures of squared exponential covariance functions," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process.*, 2014, pp. 1–6, doi: 10.1109/MLSP.2014.6958899.
[23] M. Schonlau and R. Zou, "The random forest algorithm for statistical learning," *Stata J.*, vol. 20, no. 1, pp. 3–29, 2020, doi: 10.1177/1536867x20909688.
[24] A. Gulli and S. Pal, *Deep Learning with Keras*, Birmingham: Packt, 2017.
[25] A. Richter, F. Werner, A. Cuevas, J. Schmidt, and S.W. Glunz, "Improved parameterization of auger recombination in silicon," *Energy Procedia*, vol. 27, pp. 88–94, 2012, doi: 10.1016/0038-1101(92)903257.
[26] D. B. M. Klaassen, "A unified mobility model for device simulation-I. Model equations and concentration dependence," *Solid-State Electron.*, vol. 35, no. 7, pp. 953–959, 1992, doi: 10.1016/0038-1101(92)903257.
[27] P. Singh, *Deploy Machine Learning Models to Production*. Berkeley, CA: Apress L. P., 2021.
[28] R. Stangl *et al.*, "Developing a web based PV simulation platform (targeting at machine learning combined with advanced device and process simulation to support process optimization)," in *Proc. IEEE 46th Photovolt. Specialists Conf.*, 2019, pp. 3051–3053, doi: 10.1109/PVSC40753.2019.8980687.