Distributed Solar Prediction with Wind Velocity

Justin Domke, Nick Engerer, Aditya Menon, Christfried Webers

National ICT Australia and the Australian National University

***Abstract*—The growing uptake of residential PV (photovoltaic) systems creates challenges for electricity grid management, owing to the fundamentally intermittent nature of PV production. This creates the need for PV forecasting based on a distributed network of sites, which has been an area of active research in the past few years. This paper describes a new statistical approach to PV forecasting, with two key contributions. First, we describe a "local regularisation" scheme, wherein the PV energy at a given site is only attempted to be predicted based on measurements on geographically nearby sites. Second, we describe a means of incorporating wind velocities into our prediction, which we term "wind expansion", and show that this scheme is robust to errors in specification of the velocities. Both these extensions are shown to significantly improve the accuracy of PV prediction.**

***Index Terms*—Solar energy, Solar power generation, Machine learning algorithms**

I. Introduction

A high uptake of photovoltaic (PV) systems creates instability in the electricity grid due to the intermittency of sunlight. Grid operators thus need a reliable method to estimate the contribution of distributed residential PV systems to the power grid. While there has been a large body of work focussed on PV forecasting at a single site – see e.g. Inman et al. [1] for a survey – as noted by Yang et al. [2], such methods do not provide any spatial irradiance information. As such information is invaluable to grid management, there has been a recent flurry of interest in PV forecasting techniques based on a distributed network of sites [2, 3, 4, 5, 6, 7]. These methods all focus on intra-minute to several minute data and forecast horizons, leveraging multiple ground sensors from either solar irradiance sensors or PV system power output.

In this paper, we describe a new statistical approach to short-term (5–60 minute) distributed PV prediction. Our basic model predicts the PV at a given site based on the PV at other sites, but with two important extensions. First, we enforce *local regularisation* so that the PV at a given site is only determined by the few sites that are geographically nearby, rather than all sites. Second, we describe a *wind expansion* scheme whereby wind velocities are exploited in the final predictor, such that the final predictor is robust to errors in the wind velocities. Put together, these (to our knowledge novel) ideas result in a simple but demonstrably effective distributed PV forecaster.

The idea of using wind velocities in PV forecasting is by itself not new, as several recent works have discussed the perceived importance of including this information. For example, Lonij et al. [3] demonstrated that the inclusion of the 700 mb wind from the NOAA RUC weather model was important for maximising prediction accuracy. Bosch and Kleissl [8] proposed a "triplet method" for extracting the apparent cloud velocity from a distributed sensor network, a method then used by Lipperheide et al. [6] and shown to be essential for the generation of precise forecasting across a 48MWp solar array. Furthermore, Lorenzo et al. [7] proposed and evaluated a real-time solar radiation forecasting method based on a network of 12 irradiance sensors, wherein the extracted wind velocity estimates from Weather Research & Forecasting model were essential to improvement over their previous forecasting work [3].

Each of the above studies has shown that inclusion of the cloud velocity is helpful for distributed solar forecasts. However, in this paper, we wish to go a step further, and propose a technique ("wind expansion") which demonstrates that (a) if perfectly known, the cloud motion vectors can result in improved forecasts, and (b) errors in cloud motion vectors (when estimated using an image processing heuristic) cause only mild degradation in forecast accuracy.

A. Short-term forecasting

A fairly thorough review of time-series forecasting methods at hourly, daily and monthly forecast horizons has recently been provided in Voyant et al. [9], with the majority of state of the art forecasting research moving to sub-hourly timeseries from 2013 onwards. As discussed in several papers, sub-hourly forecast horizons are "completely different" from hourly or daily timescales [9], owing to the probability distribution of persistence changing, such that conditions are much more unlikely to change between time steps.

B. Distributed versus single-site prediction

A important part of accurately forecasting the spatio-temporal variability of solar radiation fields is the deployment and use of distributed sensor networks to produce spatial solar forecasts [2]. In the area of distributed solar forecasting methods, a number of techniques appear in the literature, all of which focus on the intra-minute to a several minute data and forecast resolutions/horizons. They leverage multiple ground sensors from either solar irradiance sensors or PV system power output and generate solar forecasting techniques which leverage this joint spatial-temporal nature of the data.

[3]. A very important paper, often cited by other distributed network based solar forecasting papers, this study presented a method for fore- casting power output from 80 distributed PV systems installed in a 50 x 50 km region in Tucson, AZ which had a mean separation distance of 3km. The 15 minute interval data was normalised by the methods outlined

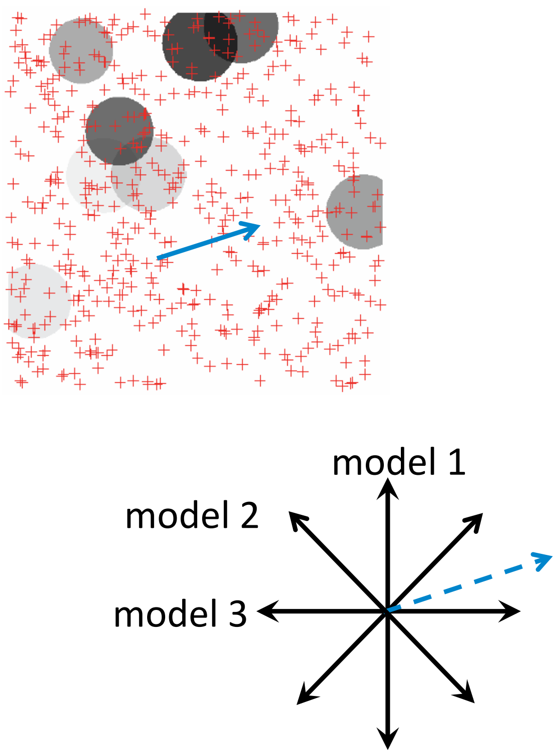


Fig. 1: The basic idea of wind expansion: If one had different predictors for a fixed mesh of wind directions, one could predict for an arbitrary wind direction by interpolating the predictions of the closest wind directions in the mesh. This paper expands the feature space so that the accuracy of this ”interpolated predictor” is itself fit at training time.

in Lonij et al. [3] and then discretized before the fields were advected forward in time according to a cloud motion vector. This motion vector was determined by four methods, including persistence, extraction of the 700 mb wind from the NOAA RUC weather model and vector extracted from the sensor network itself through choosing the wind that minimizes the RMSE of the predictions. Their approach was able to beat persistence based forecasts out to 60 minutes and demonstrated improvement over the existing satellite methods at the time (SolarAnywhere, 2012).

II. Basic Predictor

Our basic predictor is least-squared linear regression. The idea is that the power at a site 10 minutes ahead can be approximated by a linear combination (with learned weights) of the previous power measurements at all sites at the previous five time steps. However, two major changes are made to this predictor to reflect the geometric relationship between different sites, and how wind changes how this geometry is reflected in power measurements: firstly, ”local regularization” means that only sites within some fixed distance of the target can be used. If an infinite amount of training data we available, it would be best to use all sites, but given a finite amount of data, this can dramatically reduce over-fitting. Secondly, we introduce the idea of ”wind expansion” where the input measurements are transformed based on the current wind velocity. This allows the known wind direction to be incorporated in a very simple way into the predictor.

III. Simulation

For two reasons, this paper uses a somewhat crude simulated data set, as this enables experiments with unlimited numbers of sites, without the cost of manually instrumenting and measuring each of them. For the sake of disambiguating the effects of different phenomena on predictive accuracy, we present three different simulations, and ”easy”, ”medium”, and ”hard” simulation.

The ”hard” simulation can be described as follows. First, some number of sites are randomly scattered on the unit square. First, for each day, a cloud creation probability is chosen uniformly on $[0, 1]$ and an overall cloud movement uniformly chosen as (δ_x, δ_y) where δ_x and δ_y are randomly chosen from $[-0.02, 0.02]$. After each time step, representing 10 minutes, a new cloud is created with the aforementioned cloud-creation probability. A maximum radius $r_{\max}$ for clouds is chosen from a Poisson distribution. Next, a set of clouds are generated uniformly over the square, with sizes chosen as $\min(r^4, r_{\max})$ where r is chosen uniformly on $[0, 1]$. The opacity of each cloud is chosen uniformly on $[0, 1]$.

Figure 2 shows some example sites and cloud covers.

After each time step, the cloud centers are advanced by (δ_x, δ_y) and each dimension of the cloud position is corrupted by a “jigger” of a normal with standard deviation 0.005. finally, the radius is re-computed as .95 times the old radius, and .05 times a new radius randomly chosen from the same Poisson distribution. Note that this neglects the possibility that clouds at different heights might move in different directions.

For each site, an “occlusion” constant is computed by multiplying the opacity of each cloud that is overhead (Figure 3, left). Finally, the power is generated from a 10 hour sinusoid multiplied by the occlusion (Figure 3, right). All the experiments described here use one year of data for training and one year for testing.

A. Easy and Medium Simulation

These settings are slightly different in making the movement of the wind easier to predict. In the medium setting it is still somewhat random, but has a general predictable tendency, while in the easy setting the movement is completely fixed.

1) The overall wind direction is less variable. In the medium setting, the cloud movement directions δ_x and δ_y for each day are randomly chosen from $[0, 0.02]$. In the easy setting, these are fixed at $\delta_x = \delta_y = 0.01$, meaning the overall wind direction does not change.
2) In the easy setting, the random ”jigger” at each time-step to the center and radius of each cloud is removed.

IV. Local Regularization

A first experiment tested the concept of ”local regularization” in connection with the total number of available sites.

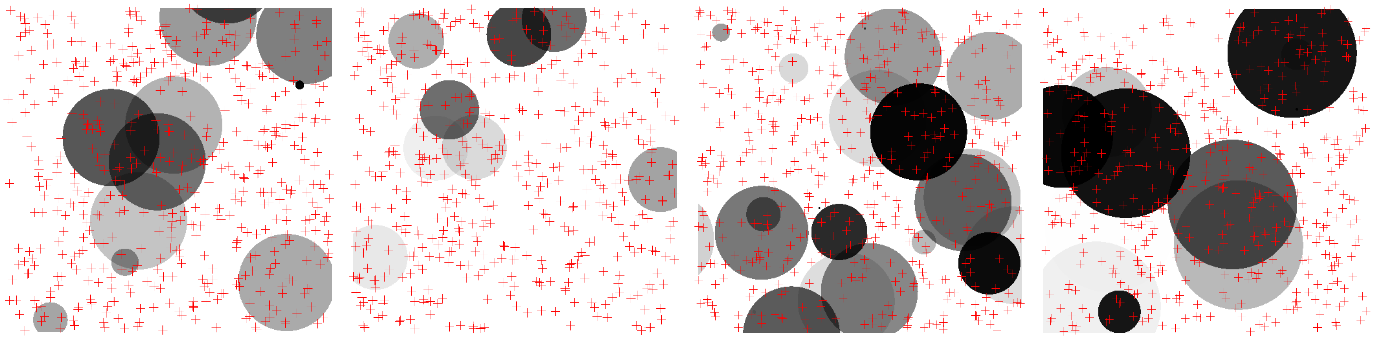


Fig. 2: Some examples of simulation with random clouds and 500 randomly scattered sites.

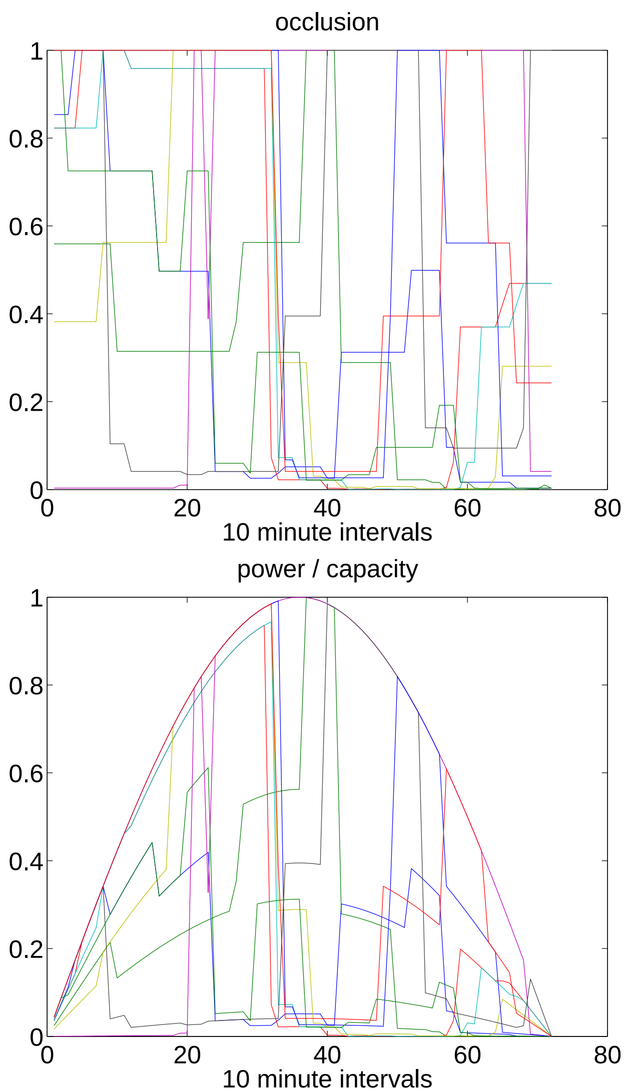


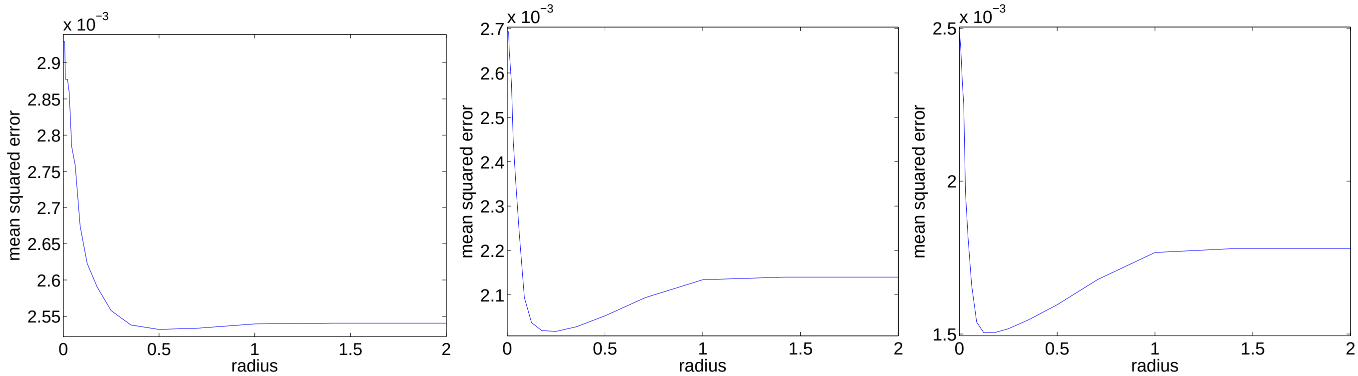
Fig. 3: Examples of occlusion tracks and simulated power (as a fraction of total site capacity) for a day. Each line corresponds to one site (10 total).

The idea is that faraway sites, since they are likely to be under different cloud cover, contain less predictive information than nearby sites. Since data is limited, it is helpful to restrict the predictor's access to extra information. Thus, we experimented with building a standard linear predictor with the single change that the predictor for one site would only have access to information from other sites within a given radius. This experiment was repeated with various numbers of total sites, and is shown in Fig. 6. The conclusions are clear: with a small number of sites, local regularization has little impact. With a large number of sites, it has a huge impact– to the degree that with 500 sites, joint prediction is no better than local prediction unless local regularization is used. Further, each of these curves has a minimum test error at a radius of around 0.1. Notice that this is much larger than the cloud movement, which is in the range on $(-0.02, -0.02)$ to $(0.02, 0.02)$. The reason for this, presumably, is that the clouds themselves are quite large, and thus somewhat distance sites can still provide valuable information. Nevertheless, with more years of training data, this step would not be necessary.

V. Wind Expansion

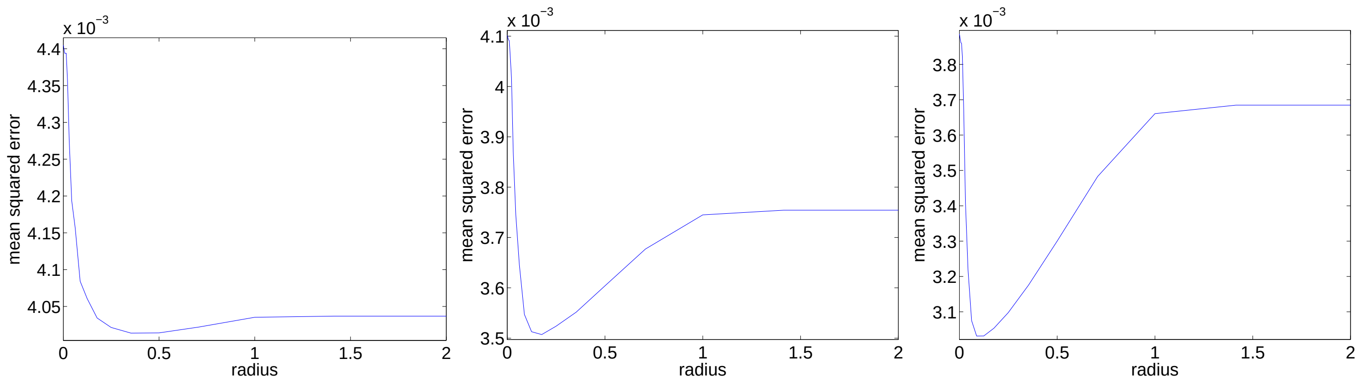
If we disregard training for a moment, and think of how a test-time predictor might take advantage of a known wind direction, it is easy to imagine a situation like the following: There are a set of different weight vectors, one for each of a mesh of possible wind directions. Given the particular known wind direction, the weights for the current situation are selected by a linear interpolation of the weights in the nearest grid points.

While training a system as described above might seem difficult, it is easy to do using the standard duality between feature-spaces and parameter-spaces. Namely, rather than interpolating weights, we expand the feature space by a factor of the number of mesh points. To create the features for a given point in time a "reverse-interpolation" is used. Namely, one calculates the weights that the point would receive from each of the neighboring mesh points. Then, each of the corresponding feature spaces is given the original features, multiplied by the corresponding weight. Training this model with a fixed set of weights and "reverse-interpolated features"



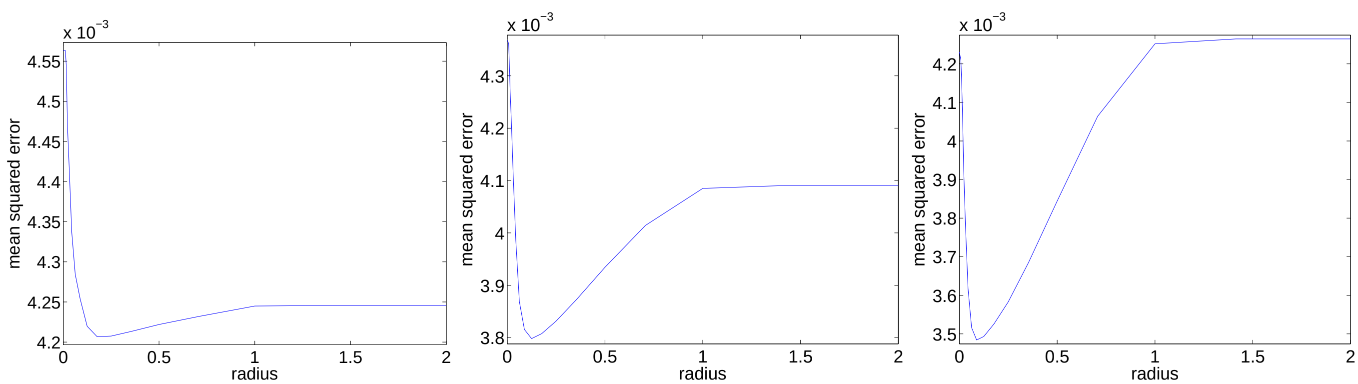
(a) 50 sites (b) 200 sites (c) 500 sites

Fig. 4: Local regularization with the ”easy” simulated data. There is a modest benefit to local regularization with many sites.



(a) 50 sites (b) 200 sites (c) 500 sites

Fig. 5: Local regularization with the ”medium” simulated data. There is a larger benefit to local regularization compared with the easy setting.



(a) 50 sites (b) 200 sites (c) 500 sites

Fig. 6: Local regularization with the ”hard” simulated data. With 500 sites, using a large radius (equivalent to skipping local regularization) has similar error to using a tiny radius (equivalent to doing single-site prediction).

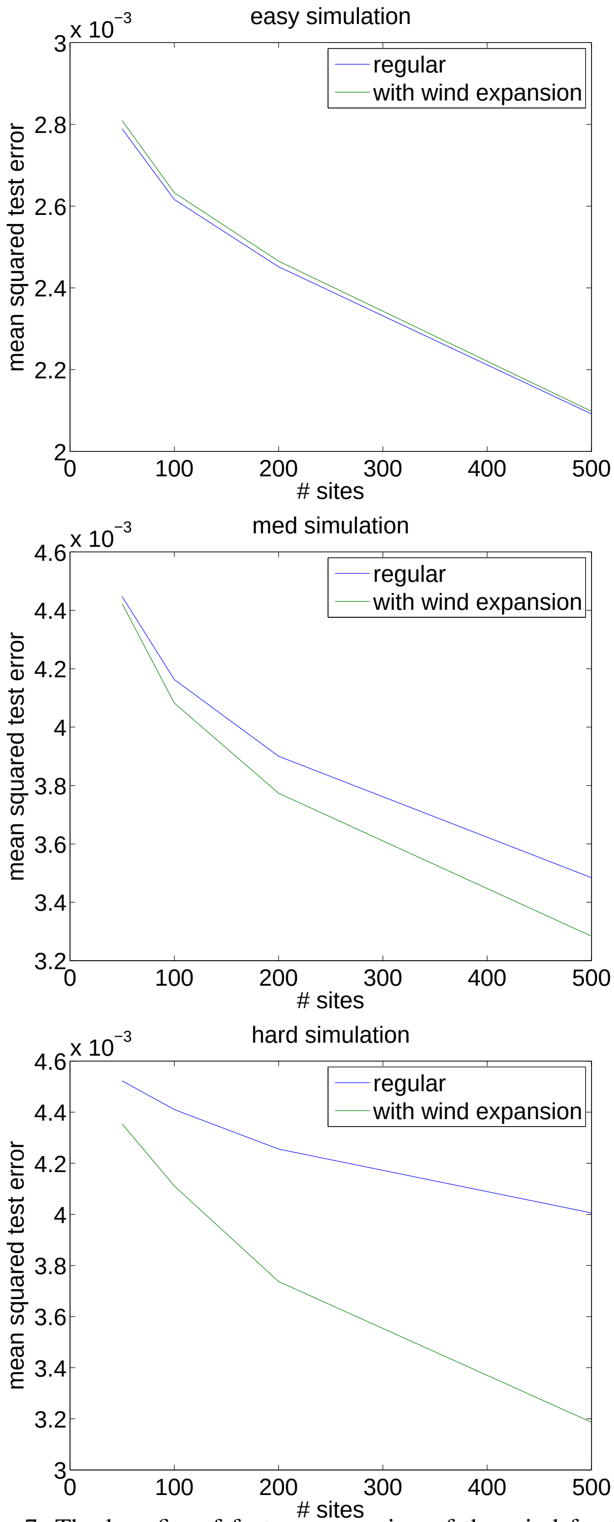


Fig. 7: The benefits of feature expansion of the wind for the three simulated settings. In the easy setting, there is no benefit, as the wind direction is fixed. There is a larger benefit in the hard setting than the medium setting, probably reflecting the more variable wind directions there.

Fig. 8: Interpolated occlusion maps 10 minutes apart with 500 simulated sites. There is significant change beyond simple translation due to irregular site locations and changes in cloud positions, sizes, and opacity.

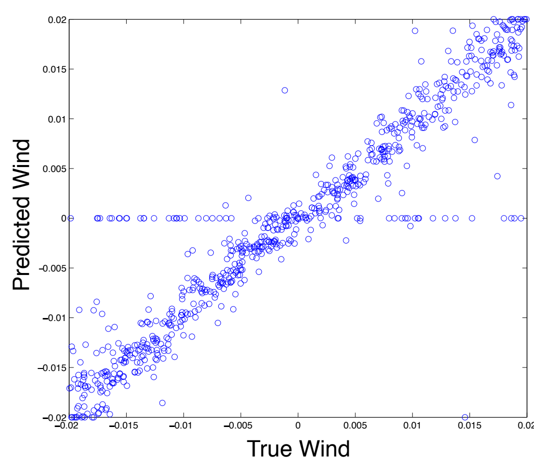


Fig. 9: Estimated vs. true wind directions. The x-axis shows one component of the true wind direction at different times, while the y-axis shows the same quantity estimated from registration of images of the type shown in Fig. 8

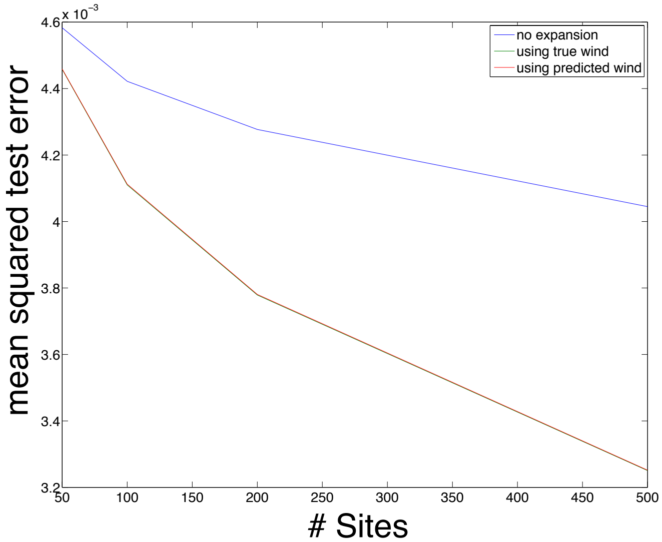


Fig. 10: Errors with estimated vs. true wind directions. The upper blue line shows the error with no wind expansion, while the (nearly identical) green and red lines show the errors with wind expansion with the true wind and predicted wind, respectively.

is exactly equivalent to training a model of the above type with an interpolated set of weights.

To test this in the simplest setting, we considered a very low-order expansion, where there are 3 grid points for the wind in each direction, for a total of 9. Thus, the feature space is expanded by a factor of nine. Because of the success of local regularization above, and because we now have an even larger feature space, this expansion is used in conjunction with the weights being constrained to a radius of 0.1, as this was nearly optimal in all the experiments above. The grid points are chosen as -0.02,0, or 0.02, since this is the range of the wind. The results are shown in Figure 7. One can see that wind expansion creates a substantial improvement, which increases with the number of sites available.

In practice, of course, one will not typically have access to the exact known wind direction. An interesting question is if this can be estimated from simulated data alone. After some experimentation, we found a very simple strategy for this. Firstly, at each time, a nonlinear interpolation is used to estimate the current occlusion map on a discretized regular grid of the image. See Figure 8 for an example. Next, a simple image transformation algorithm is used to estimate the ”shift” between the two images which is equivalent (after a change of coordinates) to an estimate of the wind. Figure 9 shows a scatter-plot of the estimated vs. true wind using this algorithm. One can see that, while far from perfect, this generally produces an estimate close to the true one. Finally, we take this estimated wind direction and use it in the final predictor in place of the true wind. This is shown in Figure 10. Remarkably, there is essentially zero difference between using the estimated wind direction and the true one. This suggests that, given a sufficiently large number of sites, the wind can be estimated accurately enough to improve the predictor. This makes sense, given that the wind is still only being used in a somewhat crude way above, via wind expansion.

Acknowledgments

NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

References

[1] R. H. Inman, H. T. Pedro, and C. F. Coimbra, “Solar forecasting methods for renewable energy integration,” *Progress in Energy and Combustion Science*, vol. 39, no. 6, pp. 535 – 576, 2013.

[2] D. Yang, C. Gu, Z. Dong, P. Jirutitijaroen, N. Chen, and W. M. Walsh, “Solar irradiance forecasting using spatial-temporal covariance structures and time-forward kriging,” *Renewable Energy*, vol. 60, pp. 235 – 245, 2013.

[3] V. P. Lonij, A. E. Brooks, A. D. Cronin, M. Leuthold, and K. Koch, “Intra-hour forecasts of solar power production using measurements from a network of irradiance sensors,” *Solar Energy*, vol. 97, pp. 58 – 66, 2013.

[4] S. Achleitner, A. Kamthe, T. Liu, and A. E. Cerpa, “SIPs: Solar irradiance prediction system,” in *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, ser. IPSN ’14. Piscataway, NJ, USA: IEEE Press, 2014, pp. 225–236.

[5] D. Yang, Z. Ye, L. H. I. Lim, and Z. Dong, “Very short term irradiance forecasting using the lasso,” *Solar Energy*, vol. 114, pp. 314 – 326, 2015.

[6] M. Lipperheide, J. Bosch, and J. Kleissl, “Embedded nowcasting method using cloud speed persistence for a photovoltaic power plant,” *Solar Energy*, vol. 112, pp. 232 – 238, 2015.

[7] A. T. Lorenzo, W. F. Holmgren, and A. D. Cronin, “Irradiance forecasts based on an irradiance monitoring network, cloud motion, and spatial averaging,” *Solar Energy*, vol. 122, pp. 1158 – 1169, 2015.

[8] J. Bosch and J. Kleissl, “Cloud motion vectors from a network of ground sensors in a solar power plant,” *Solar Energy*, vol. 95, pp. 13 – 20, 2013.

[9] C. Voyant, C. Paoli, M. Muselli, and M.-L. Nivet, “Multi-horizon solar radiation forecasting for mediterranean locations using time series models,” *Renewable and Sustainable Energy Reviews*, vol. 28, pp. 44 – 52, 2013.