TESLA: Taylor Expanded Solar Analog Forecasting

Bengu Ozge Akyurek*, Alper Sinan Akyurek†, Jan Kleissl* and Tajana Šimunić Rosing†
*Mechanical and Aerospace Engineering
University of California - San Diego
Email: {bakyurek,jkleissl}@ucsd.edu
†Electrical and Computer Engineering
University of California - San Diego
Email: {aakyurek,tajana}@ucsd.edu

***Abstract*—With the increasing penetration of renewable energy resources within the Smart Grid, solar forecasting has become an important problem for hour-ahead and day-ahead planning. Within this work, we analyze the Analog Forecast method family, which uses past observations to improve the forecast product. We first show that the frequently used euclidean distance metric has drawbacks and leads to poor performance relatively. In this paper, we introduce a new method, TESLA forecasting, which is very fast and light, and we show through case studies that we can beat the persistence method, a state of the art comparison method, by up-to 50% in terms of root mean square error to give an accurate forecasting result. An extension is also provided to improve the forecast accuracy by decreasing the forecast horizon.**

I. Introduction

History repeats itself. Weather is a continuous, data-intensive, multidimensional, dynamic and chaotic process, and these properties make weather forecasting a formidable challenge. With the increasing percentage of renewable energy penetration within the Smart Grid, forecasting the weather accurately gained even more importance. Even now, a group of Smart Grid control algorithms, battery optimization solutions [1], day ahead energy market negotiations and residential energy management systems [2] already rely on the availability of an accurate forecast. High errors in generation forecasts have the danger of disturbing the supply-demand stability within the Smart Grid, which will have to be compensated by expensive generators or in the worst case may even lead to frequency drop and instabilities.

Weather forecasts provide critical information about future weather. There are a wide range of techniques involved in weather forecasting from basic approaches to highly complex computerized models [3]. It is difficult to obtain an accurate result from the weather and solar predictions. Accurate forecasting of solar irradiance is essential for the efficient operation of solar thermal power plants, energy markets, and the widespread implementation of solar photovoltaic technology. Numerical weather prediction (NWP) is generally the most accurate tool for forecasting solar irradiation several hours in advance [4]. The techniques used in solar forecasting can be categorized as dynamical and empirical methods. Furthermore, NWPs provide another alternative to a national or global scale ground based monitoring network [5]. NWP models provide a comprehensive and physically-based state-of-the-art description of the atmosphere and its interactions with the Earth surface [6]. But, these methods are very computation intensive and require both time and computation resources. In order to enhance the forecast accuracy, there are refining techniques [7].

Most weather prediction systems use a combination of empirical and dynamical techniques. However, a little attention has been paid to the use of artificial neural networks (ANN) in weather forecasting [3], [8]. Since the late 1990s ANNs have seen increased application in the field of solar forecasting [5].

ANNs provide a methodology for solving many types of non-linear problems which are difficult to solve by traditional techniques [9]. Furthermore, ANN modeling offers improved non-linear approximation performance and provides an alternative approach to physical modeling for irradiance data when enough historical data is available.

History repeats itself. Another family of methods is the analog method family. It relies on this fact that tomorrow has already happened in the past. In [10], the authors also show that the different applications of analog method usage in their study. They used different types of k-methods in their studies, which gives a good idea on the accuracy and the errors of the methods. In addition to this Hacker [11] also compares the different types of analogue approaches in their studies by indicating the inclusion of model diversity showing an improvement in terms of reliability and the statistical consistency in an analogue method approach study. Analog methods have been applied and tested for forecasting increasingly. Abdel-Aal shows the effect of using different training sets to train a network system [12].

In this work, we analyze the fundamental pieces of the analog forecast method family and show that the distance, which describes the similarity between two analogues is very important for a good forecast. First, we propose an extension to the euclidean distance based analog method, then generalize the idea to construct a new method called Taylor Expanded Solar Analog (TESLA) forecasting. We show through case studies that we can even beat the persistence method by up-to 50% in terms of solar irradiance root mean square error (RMSE).

The rest of this paper is organized as follows. Section II analyzes the most commonly used Euclidean distance and shows why the forecast will not perform well with this metric and proposes an extension to improve it. Section III describes the working principle of our algorithm TESLA. Section IV shows that our algorithm performs very well compared to the state-of-the-art methods on case studies.

II. METHODOLOGY

Before we begin with the proposed method, we start by explaining the motivation for searching for a better forecasting algorithm by going over the drawbacks of some of the algorithms frequently used in the literature.

A. Euclidean Distance Analog Method

The analog forecasting method relies upon the fact that the history consists of recurrences. In other words, the future may have already happened in the past. In order to establish a connection between the future and the past, we first need a rough forecast product of the future and a distance of this forecast to the multiple forecast points in the past. This metric will describe how much the forecasted day is **similar** to the days that have happened in the past.

The forecasts produced in the past are grouped under the name of *ensembles*. Each ensemble has also an *observation* associated with it, which is the solar irradiance observed at the time of the ensemble, measured by weather stations. The euclidean distance analog method simply measures the euclidean distance between the forecast and each ensemble and weighs the observations inversely proportional to the distance. We can write this algebraically. First, we need to define the variable names that are going to be used throughout this paper.

- $e_{i,j}$: The forecast product has multiple outputs, typically forecasting the states of the weather like temperature at various atmosphere heights, relative humidity or wind speeds. This variable is the j^{th} variable output of the i^{th} hourly forecast ensemble.
- o_i: The observed/measured solar irradiance associated with the i^{th} forecast ensemble.
- $f_{k,j}$: The f variable defines the rough forecast of the desired future, thus this variable defines the j^{th} variable output of the k^{th} hour future forecast product.

Figure 1 shows an example construction to clarify the concept and the timing of the variables.

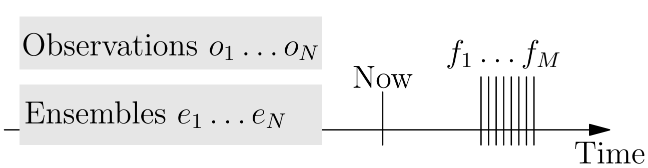

Fig. 1. An example timeline showing the construction of the Analog Forecasting method family.

Using the defined variables, we can define the euclidean distance method algebraically. The euclidean distance in an N_j dimension universe is defined as:

$$d(x, y) = \sqrt{\sum_{j=1}^{N_j} (x_j - y_j)^2} \tag{1}$$

Applying this distance to our i^{th} ensemble and k^{th} forecast:

$$d_{i,k} = \sqrt{\sum_{j=1}^{N_j} (e_{i,j} - f_{k,j})^2} \tag{2}$$

The analog forecast output can be defined by weighing the observations inversely proportional to the distance:

$$a_k = \frac{\sum_{i=1}^{N_i} \frac{o_i}{d_{i,k}}}{\sum_{i=1}^{N_i} \frac{1}{d_{i,k}}} \tag{3}$$

Note that this method has its drawbacks. First of all, the method relies on the fact that if the distance of two forecast products is small then their observations should be close to each other, in other words they will be similar days in terms of weather.

To check how well the euclidean distance metric performs, we have constructed an ensemble set of 16343 hours (roughly 15 months). The set is obtained from NOMADS, North American Mesoscale (NAM) [13] forecast data, which consists of 36 hour daily forecasts. We have selected 38 variables from the forecasts for distance calculation. We have sorted all ensembles and corresponding observations in ascending order of observations. Then we calculated the distance of all ensembles to three selected ensembles, 2000, 10000 and 13000 corresponding to a night and two mid day indices. The resulting plots are given in Figure 2.

In the ideal case, we would expect the ensembles close to the selected ensembles to have a small distance, since their observations are close. As the indices go far from the selected ensembles, the distance metric should increase since the similarity between observations will be lost completely. The figure shows that the real case is very far from the expected ideal case. There is a big band of noise in the figures, furthermore the expected increase in distance is not observed around the selected indices. This non-ideality of the distance metric causes a big problem on the accuracy of the forecasts, as will be shown in the performance section.

Another drawback of this method is that it uses a linear combination of the variables to calculate the distance metric, but in reality this may not be the case. We would only have a linear approximation of the ideal distance.

A final remark on the method is that the weight of each parameter on the distance metric is assumed to be the same, creating a perfect hyper-sphere. Different parameters may have different weighted effects on the distance, creating a hyper-ellipse rather than a hyper-sphere. The application of this idea is explained in the next section as an extension to the euclidean distance method.

B. Weighted Euclidean Distance Analog Method

In the previous section, we have shown that the distance metric to measure the similarities between ensembles is not a completely reliable metric. In this section, we propose an extension to the euclidean distance analog method by introducing linear weights to incorporate different effects of

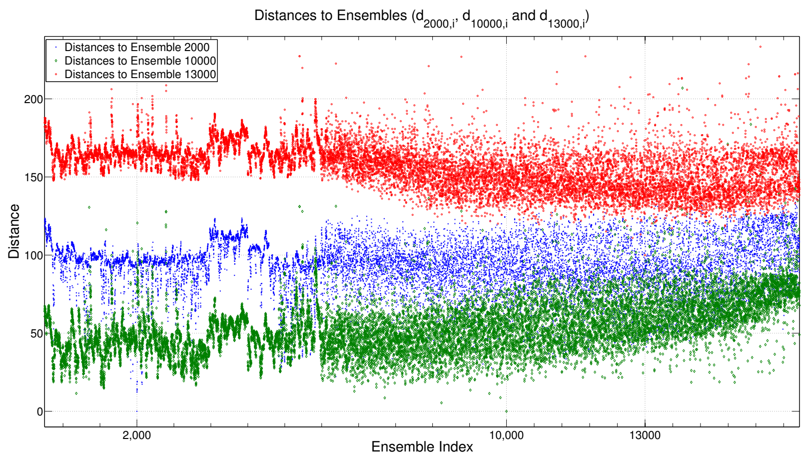

Fig. 2. The euclidean distance of all sorted ensembles to the Ensemble 2000 (blue), Ensemble 10000 (green) and Ensemble 13000 (red).

forecast variables into the distance metric. We can show this algebraically as:

$$dW_{i,k} := \sqrt{\sum_{j=1}^{N_j} w_j \left(e_{i,j} - f_{k,j}\right)^2} \tag{4}$$

This new parameter introduces the problem of determining its value. In order to find the optimal weight values, we need to train the system with known outcomes and optimize the system to best fit the expected outcomes. In order to formulate the optimization problem, we need to introduce two more variables.

- $t_{k,j}$: We use a set of ensembles and their associated observations to train the system weights. This variable defines the j^{th} parameter of the k^{th} training ensemble.
- γ_k: This variable defines the observation associated with the k^{th} training ensemble.

Given that we have our training ensembles and their associated observations, we can define our optimization problem. The main objective is to maximize the accuracy of our forecasts. We define the accuracy of multiple forecasts as the Root Mean Square Error (RMSE). The RMSE for the training ensembles is given in the following equation.

$$RMSE = \sqrt{\frac{1}{N_k}\sum_{k=1}^{N_k}\left(\frac{\sum_{i=1}^{N_i}\frac{o_i}{dWT_{i,k}}}{\sum_{i=1}^{N_i}\frac{1}{dWT_{i,k}}} - \gamma_k\right)^2} \tag{5}$$

where

$$dWT_{i,k} := \sqrt{\sum_{j=1}^{N_j} w_j \left(e_{i,j} - t_{k,j}\right)^2} \tag{6}$$

This equation is optimized using the Optimization Toolbox in MATLAB. Using 14000 training hours we have obtained the optimal weights and tested our new distance metric on the example ensembles from the previous section. The distances are shown in Figure 3.

It can be clearly seen that the introduction of parameter weights has improved the shapes of the distance metrics to the expected ideal case. For Ensemble 13000, it can be seen that the distance to the closer points is small compared to the farther ensembles, constructing the convex shape that was desired. Although the general trends of the distances have improved, there is still too much noise in the system that will lead to errors if not handled. In the performance section, we will show that the weighted distance method performs better than its uniform counterpart. Next, we will define even a better solution, the main method presented in this paper, in the next section.

III. TESLA: Taylor Expanded SoLar Analog Forecasting

In the previous section, we have shown that the distance metric showing the similarity between ensembles has problems and needs improvement, because the similarity is the heart of the analog forecasting method family. A second observation that we made in the previous section is that the similarity is calculated as a linear combination of the parameters, which in real life may not be the case. To better address these problems, we have changed our perspective fundamentally. Instead of using a *distance* metric, we introduce a new *similarity* metric. This metric is constructed as a function of two vectors, representing the two ensembles that we are comparing for similarity and outputs a similarity value. This can be formulated as:

$$s_{i,k} = S(\mathbf{e_i}, \mathbf{f_k}) \tag{7}$$

We aren't assuming anything regarding how this similarity function should be. Instead, we write the Taylor Expansion

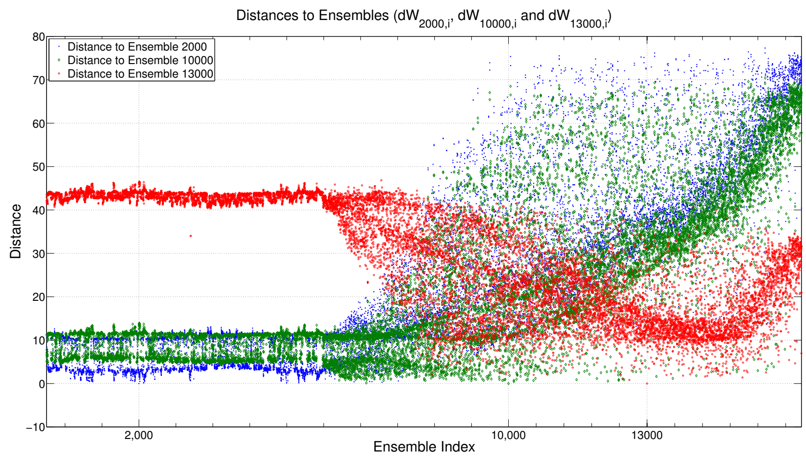

Fig. 3. The weighted euclidean distance of all sorted ensembles to the Ensemble 2000 (blue), Ensemble 10000 (green) and Ensemble 13000 (red).

of the similarity function around $(\mathbf{0}, \mathbf{0})$.

$$\begin{aligned} S(\mathbf{x}, \mathbf{y}) &= S(\mathbf{0}, \mathbf{0}) + \sum_{j=1}^{N_j} \partial_{x_j} S(\mathbf{0}, \mathbf{0}) x_j + \sum_{j=1}^{N_j} \partial_{y_j} S(\mathbf{0}, \mathbf{0}) y_j \\ &+ \frac{1}{2!} \sum_{j1=1}^{N_j} \sum_{j2=1}^{N_j} \partial_{x_{j1}, x_{j2}} S(\mathbf{0}, \mathbf{0}) x_{j1} x_{j2} \\ &+ \sum_{j1=1}^{N_j} \sum_{j2=1}^{N_j} \partial_{x_{j1}, y_{j2}} S(\mathbf{0}, \mathbf{0}) x_{j1} y_{j2} \\ &+ \frac{1}{2!} \sum_{j1=1}^{N_j} \sum_{j2=1}^{N_j} \partial_{y_{j1}, y_{j2}} S(\mathbf{0}, \mathbf{0}) y_{j1} y_{j2} + \ldots \end{aligned}$$

Note that within this expansion, we don't know any of the expansion constants. We can denote the unknowns as a_i, such that;

$$a_1 = S(\mathbf{0}, \mathbf{0}),\ a_{2,j} = \partial_{x_j} S(\mathbf{0}, \mathbf{0}),\ a_{3,j} \partial_{y_j} S(\mathbf{0}, \mathbf{0}), \ldots$$

This expression can be represented in matrix form. Representing all the unknown constants as the $\mathbf{A}$ vector and concatenating the variables into a single vector ψ, the equality becomes:

$$S(\mathbf{x}, \mathbf{y}) = \mathbf{A}^T \cdot \psi \tag{8}$$

We need to find a way to obtain the $\mathbf{A}$ parameters. We again use a forecast set for training purposes.

The similarity between an ensemble and a training forecast can be found as:

$$S(\mathbf{e_i}, \mathbf{t_k}) := s_{i,k} = \mathbf{A}^T \cdot \psi(\mathbf{e_i}, \mathbf{t_k})^T = \mathbf{A}^T \cdot \psi_{i,k} \tag{9}$$

Concatenating the similarity metrics for all ensembles horizontally:

$$\mathbf{S}(\mathbf{e}, \mathbf{t_k}) := \mathbf{s_k} = \mathbf{A}^T \cdot (\psi_{1,k} \quad \cdots \quad \psi_{N_i,k}) := \mathbf{A}^T \cdot \mathbf{B_k} \tag{10}$$

Using the similarity metric, we define our analog forecast result as:

$$a_k = \sum_{i=1}^{N_i} o_i s_{i,k} = \mathbf{s_k} \cdot \mathbf{o} = \mathbf{A}^T \cdot \mathbf{B_k} \cdot \mathbf{o} \tag{11}$$

The final step is to define a vector $\mathbf{M_k} = \mathbf{B_k} \cdot \mathbf{o}$ and concatenating the vectors horizontally for all k values, converting it into a matrix $\mathbf{M}$. The forecast result vector is then simply found as:

$$\mathbf{a} = \mathbf{A}^T \cdot \mathbf{M} \tag{12}$$

We want in the ideal case $\mathbf{a} = \gamma$ to have an error-free forecast. In most cases the rank of $\mathbf{M}$ matrix is less than the size of the training set, N_t. This means that we have an under-defined system of equations. Since we are trying to minimize the RMSE, we can solve this system by Least Mean Squares Estimation (LMSE) to get the Taylor Expansion parameters.

After an initial training, the system will have learned the Taylor Expansion terms and use them to calculate the similarity metrics and give the TESLA Forecast result. Note that, this method is already a super-set of both euclidean distance methods explained in the previous section. Furthermore, if the order of the Taylor Expansion is selected more than 1, the non-linear effects are also being added into the forecast, providing a better forecast.

Before moving into the performance section, an extension is provided in the next section to further improve the forecast results by trading off accuracy with the forecast horizon.

A. TESLA Forecasting with Moving Horizon Feedback Extension

TESLA method in the previous section uses a training dataset to determine its Taylor Expansion terms. Any new ensemble or observations during normal operation is not used, where it could have been used as an additional feedback parameter to increase the performance of the future forecasts.

An extension idea to TESLA is to use the N observations prior to the current forecast ensemble, as additional parameters to the ensemble parameters. Although it will be shown that the increased number of parameters by adding additional observations improves the performance, the trade-off that we are sacrificing is the forecast horizon. The forecast horizon of the TESLA method is upper-limited by the forecast horizon of the ensembles, denoted hereby by H. At any point in time, the closest observation that we have is the previous interval. The forecast interval that we are going to add our latest observation as an additional parameter, will also limit our forecast horizon. In other words, if we define the time between our latest observation and the forecast interval that we are going to add the observation to as the *delay*, denoted as D, our forecast horizon decreases to D. For an ensemble at t, N observations from time $(t - D)$ to $(t - D - N + 1)$ are added as the N additional parameters.. These concepts are described in Figure 4.

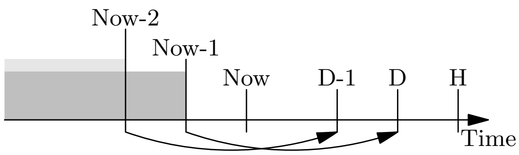

Fig. 4. For a selected delay D, the closest N observations are used as additional parameters for the D^{th} forecast. The selected window is moved step-by-step and added to the previous forecast parameters.

The smaller the value of D, the better performance we will have. This extension allows the user to determine its own forecast horizon, D, according to the error requirements. Furthermore, we can also run this method H times and varying the delay from 1 to H. By selecting the last forecast at each run, we would both get the improvement without sacrificing the forecast horizon.

IV. Performance

In order to test the performance of our solution, we have performed multiple case studies. This section describes the datasets that we have used and compares the performance of TESLA method with different methods from the literature.

A. Datasets

To construct our ensemble, training dataset and comparison datasets, we have used the 12 km, hourly NAM forecasts from September 2010 to January 2012, accounting for more than 15 months. The working site has been selected as San Diego, particularly the University of California, San Diego campus. The observation information has been used from Solar Anywhere data [14].

The NAM forecast has a 36 hour forecast horizon. We have extracted 38 parameters to be used within the ensembles. The parameters are Global Horizontal Irradiance, Planetary Boundary Layer height, surface heat flux, latent heat flux, total columnar cloud cover, dew point, surface temperature and pressure. In addition, the height, temperature, relative humidity, x and y components of wind speed have been used for barometric heights of 925, 850, 700, 500 and 200hPa, which correspond to the heights contours with the given pressure values, constituting the 38 parameters in each ensemble, that are believed to have an effect on the solar forecast result physically.

B. Weighted Euclidean Distance Performance

In order to test the performance of the weighted euclidean distance method, we have selected various sizes for our training set to understand its impact on the forecast performance. The error is calculated as the difference between the forecast

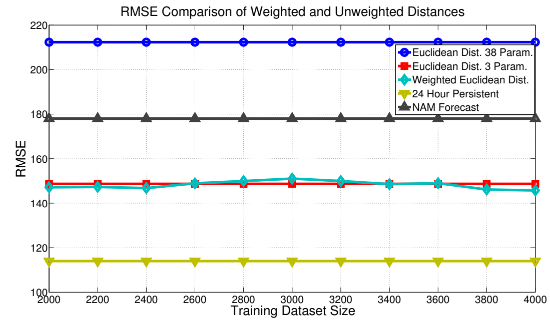

Fig. 5. RMSE(W/m^2) comparison of weighted Euclidean distance forecast and other various methods with increasing training size. The 3-parameters are the highest correlation parameters with the observation. 38 parameter distance definitely has a poor performance as expected from Section II.

product and the real solar irradiance observed on that hour shown in these equations:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(forecast_i - observation_i)^2}$$

We have selected a separated training set and comparison set. This means that the weights are determined through the training set and then the error metrics are calculated over the non-overlapping comparison set. The size of the comparison set is selected as 8300 hours.

The results are compared to the NAM and 24 hour persistence forecast results. The RMSE results are shown in Figure 5. It can be clearly seen that the euclidean distance with 38 parameters performs the worst as expected from Section II. The three parameters are selected as the highest correlation parameters of the ensembles with the observations, which performs very close to the weighted euclidean distance case. All methods still need improvement as they are much worse compared to the 24 hour persistence forecast method.

C. TESLA Forecasting Performance

We have compared TESLA against three methods. The first method is one of the state of the art methods, the analog method using Delle-Monache [15] distance as the similarity metric. The second method is also another state of the art method, the Persistence Forecast method. We have compared against both 24 hour and 1 hour persistence methods. Note that the forecast horizon of the 1 hour persistence is 1 hour. The third method for comparison is the unmodified NAM forecast. In order to make the comparison under same conditions, the 36-hour horizon of NAM forecasts are cropped to 24 hours.

Two cases are considered to test the performance of TESLA. The first case uses a training size of 450 days. The size of the overlapping comparison set is varied from 20 to 460 days. The second case uses two completely separate sets for comparison and training/ensemble. 267 days are used for comparison. The training set size is varied from 20 to 200 days. The TESLA forecast parameters are selected as: First and second order Taylor expansion and First order Taylor expansion with the Moving Horizon Feedback extension with 24 previous observations: $D = 24, D = 1$ and delay varied from 1 to 24 hours and best forecasts are combined for 24 hour forecast horizon.

The results of the first case is shown in Figure 6. The figure shows that all TESLA methods have a better RMSE than the NAM and 24 persistence method. The second order expansion and all extension results have lower RMSEs than the Delle-Monache and 1 hour persistence methods. When the forecast horizon is decreased to 1 hour ahead, we can have RMSEs as low as $50 W/m^2$.

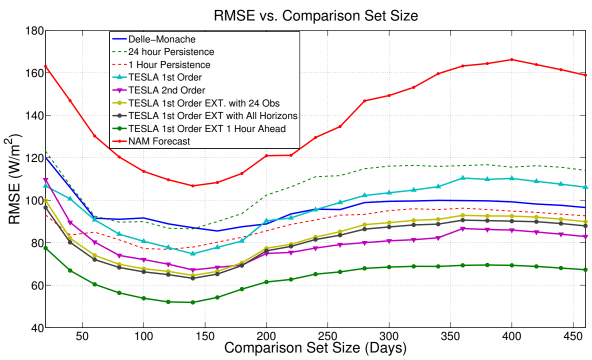

Fig. 6. Comparison of multiple methods with TESLA. The initial decrease of RMSE is due to the fact that training size also increases with comparison set size and at least 60-80 days are required to settle training.

The second case results are shown in Figure 7. TESLA requires training to construct its expansion constants. The figure shows that in order to get a good forecast, we require 60-80 days of training data. When enough training is used, TESLA performs very close to the 1 hour persistence, while maintaining the 24 hour horizon. If the horizon is decreased to 1 hour ahead, TESLA performs 25% better than the 1 hour persistence method and 50% better than the 24 hour persistence.

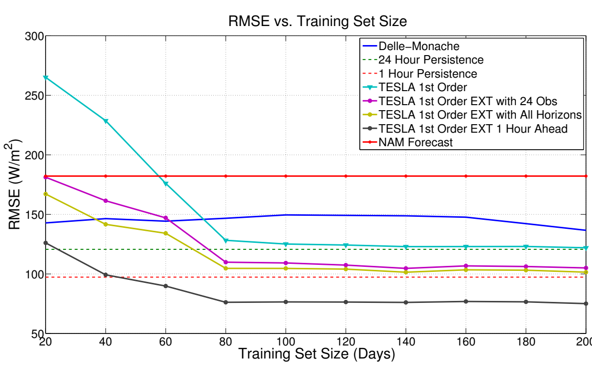

Fig. 7. Comparison of methods with TESLA under different training sizes. A minimum number of 60-80 days of training is required to get normal results.

D. Computational Complexity

TESLA computation consists of two stages, the initial training and the actual computation. Training stage, using least squares estimation has a complexity of $O(C^2N)$, where C is the number of parameters and N is the training size in our case. This stage is only performed once. The actual forecasting is a matrix multiplication with a linear complexity of $O(C)$.

V. Conclusion

Stability and load/supply are two of the most important aspects in SmartGrids. A group of grid control algorithms, day ahead negotiation markets, home automation systems require an accurate input of solar forecasting. In this paper, we have developed a new method called TESLA forecasting, which can do a 1 year forecast calculation in seconds. With case studies, we have shown that our method has a better RMSE than the state of the art forecast method. Furthermore, we have provided a Moving Horizon Feedback extension to give the ability to change the forecast horizon to obtain even more accurate results.

Acknowledgment

This work was supported in part by TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA.

References

[1] A. Akyurek, B. Torre, and T. Rosing, "Eco-dac energy control over divide and control," in *Smart Grid Communications (SmartGridComm), 2013 IEEE International Conference on*, Oct 2013, pp. 666–671.

[2] J. Venkatesh, B. Aksanli, J.-C. Junqua, P. Morin, and T. Rosing, "Homesim: Comprehensive, smart, residential electrical energy simulation and scheduling," in *Green Computing Conference (IGCC), 2013 International*, June 2013, pp. 1–8.

[3] I. Maqsood, M. R. Khan, and A. Abraham, "Intelligent weather monitoring systems using connectionist models," *NEURAL PARALLEL AND SCIENTIFIC COMPUTATIIONS*, vol. 10, no. 2, pp. 157–178, 2002.

[4] P. Mathiesen and J. Kleissl, "Evaluation of numerical weather prediction for intra-day solar forecasting in the continental united states," *Solar Energy*, vol. 85, no. 5, pp. 967–977, 2011.

[5] 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.

[6] J. Ruiz-Arias, C. Gueymard, J. Dudhia, and D. Pozo-Vázquez, "Improvement of the weather research and forecasting (wrf) model for solar resource assessments and forecasts under clear skies," *Proc. Amer. Sol. Energy Soc., Denver, USA*, 2012.

[7] A. Kaur, H. T. Pedro, and C. F. Coimbra, "Ensemble re-forecasting methods for enhanced power load prediction," *Energy Conversion and Management*, vol. 80, pp. 582–590, 2014.

[8] R. J. Kuligowski and A. P. Barros, "Localized precipitation forecasts from a numerical weather prediction model using artificial neural networks." *Weather & Forecasting*, vol. 13, no. 4, 1998.

[9] I. Maqsood, M. R. Khan, and A. Abraham, "An ensemble of neural networks for weather forecasting," *Neural Computing & Applications*, vol. 13, no. 2, pp. 112–122, 2004.

[10] M. Bannayan and G. Hoogenboom, "Weather analogue: A tool for real-time prediction of daily weather data realizations based on a modified k-nearest neighbor approach," *Environmental Modelling & Software*, vol. 23, no. 6, pp. 703–713, 2008.

[11] J. P. Hacker and D. L. Rife, "A practical approach to sequential estimation of systematic error on near-surface mesoscale grids." *Weather & Forecasting*, vol. 22, no. 6, 2007.

[12] R. Abdel-Aal, "Improving electric load forecasts using network committees," *Electric Power Systems Research*, vol. 74, no. 1, pp. 83–94, 2005.

[13] G. K. Rutledge, J. Alpert, and W. Ebisuzaki, "Nomads: A climate and weather model archive at the national oceanic and atmospheric administration," *Bulletin of the American Meteorological Society*, vol. 87, no. 3, pp. 327–341, 2014/05/11 2006.

[14] C. P. Research. Solaranywhere. [Online]. Available: solaranywhere.com

[15] L. Delle Monache, T. Nipen, Y. Liu, G. Roux, and R. Stull, "Kalman filter and analog schemes to postprocess numerical weather predictions." *Monthly Weather Review*, vol. 139, no. 11, 2011.