Short to Medium Range Time Series Prediction of Solar Irradiance Using an Echo State Network

Stephen M. Ruffing, *Member, IEEE*, and Ganesh K. Venayagamoorthy, *Senior Member, IEEE*
Real-Time Power and Intelligent Systems Laboratory
Missouri University of Science and Technology
Rolla, MO, USA
gkumar@ieee.org

***Abstract*— An Echo State Network (ESN) can make multi-step predictions since it can process temporal information without the training difficulties encountered by conventional recurrent neural networks. An ESN is applied in this paper to make multi-step predictions of solar irradiance, 30 minutes to 270 minutes into the future. The ESN is trained and tested using two performance metrics (correlation coefficient and mean squared error) on meteorological and solar data recorded at the National Renewable Energy Laboratory Solar Radiation Research Laboratory in Golden, Colorado. When feedback of target outputs is utilized, an improvement is seen for the first performance metric, while no significant change is seen for the second performance metric. Additionally, accuracy is observed to diminish significantly as the time horizon for the predictions increases.**

***Index Terms*—Echo State Network (ESN), time series multi-step prediction, solar irradiance**

I. Introduction

THE need for a shift from fossil fuel to renewable energy sources has become increasingly apparent in recent years. Although other renewable energy sources, such as hydroelectric and wind energy are currently utilized to a greater extent [1], the use of solar energy is increasing significantly [2]. The electrical power output of a photovoltaic cell, of course, hinges on the magnitude of solar irradiance incident on it and, therefore, making accurate predictions of this quantity is of critical importance. Artificial Neural Networks (ANNs) are noted for their abilities as universal function approximators [3] and, therefore, have been utilized extensively for this purpose. Previous research with ANNs [4-7] has focused on daily or seasonal predictions at locations with unknown solar irradiance, but known meteorological data, through the use of an ANN trained at another location with both meteorological and solar irradiance measurements. Solar irradiance is also difficult to predict at a given location on a time scale of minutes or hours due to cloud cover. As electricity produced from solar energy becomes a larger percentage of total generation, these short to medium time range fluctuations have an increasing impact on the electricity supply and its cost. This motivates utilities to move away from the practice of purchasing all the electricity produced by a solar or wind farm at a flat rate and, instead to make purchasing decisions on the open energy market based on short range supply predictions [8, 9]. Previous research involving these predictions for solar irradiance has relied on regional weather forecasts [8] or autoregressive moving average (ARMA) models [10].

In this study, an ANN is used to make these short to medium range predictions for solar irradiance. For time series data predictions such as these, ANN structures that incorporate recurrence have been shown to be the most accurate [3]. Conventional recurrent networks have input, hidden, and output layers, as is seen in the multi-layer perceptron (MLP) structure. The methods for training these conventional recurrent neural networks, however, are, in the best case, relatively slow and, in the worst case, prone to not converge [11]. The Echo State Network (ESN) is a form of recurrent neural network (RNN) that was developed by Jaeger in 2001 in order to overcome these training difficulties [11] and, therefore, is the network structure chosen for this study.

Previous research has shown extraterrestrial irradiance, relative humidity, temperature, barometric pressure, and wind speed to be among the most useful data for making solar irradiance predictions [4-7] and, therefore, values for each of these parameters were used as inputs to the ESN. Extraterrestrial irradiance is the amount of global horizontal irradiance that a location on the Earth's surface would receive if there were no atmosphere or clouds (i.e., directly above that location in outer space). The term global indicates the total solar irradiance, which is the sum of its direct, diffuse, and ground reflected components. The goal of this previous research was to make daily or seasonal predictions in locations where no instrumentation for measuring solar irradiance existed. Due to this factor, the present value of global horizontal irradiance was excluded as an input to the ANN. In this study, it is assumed this instrumentation is available at the location where the predictions are being made. In order to capitalize on this information, this parameter was also used as an input to the ESN.

These inputs to the ESN used in the training and test sets were taken from data recorded in one minute intervals using instruments located at the National Renewable Energy Lab (NREL)'s Solar Radiation Research Laboratory [12] in Golden, Colorado. Each data point represents the mean of readings taken every 3 seconds during that minute. The training data was used to optimize the synaptic weights of the

This work was supported by the National Science Foundation, USA under Grant EFRI #0836017.

978-1-4244-5098-5/09/$26.00 ©2009 IEEE

ESN in order to maximize the average accuracy of predictions for global horizontal irradiance at times of 30, 60, 90,….,270 minutes into the future for the test data set.

The remainder of this paper is organized as follows: A detailed discussion of the ESN architecture is given in section II; in section III, the training and testing method is explained; the results are provided in section IV, and conclusions of this work are provided in section V.

II. ECHO STATE NETWORK

A defining characteristic of RNNs that distinguish them from feedforward ANNs is that, when driven by an input signal, an RNN retains in its internal state a nonlinear transformation of the input history. This dynamical memory allows an RNN to process temporal information [13]. An ESN retains this state within its dynamical reservoir, which is depicted in Fig. 1 along with the rest of the ESN network structure. The dynamical reservoir is a large collection of sparsely connected neurons. Each output of the ESN is a linear combination of the reservoir neuron states and the inputs. This ability to "echo" its internal states in its output gives this network its name. The connections between each of the reservoir neurons and each of the outputs are referred to as either output or readout weights [14]. These readout weights and the weights connecting the inputs and outputs are adapted during training to provide ESN outputs that best approximate measured target values. The ESN also possesses input weights that connect each of the inputs to each of the reservoir neurons and reservoir weights that interconnect the reservoir neurons. These two sets of weights are not adapted during training but, instead, are maintained at fixed values. The fixed values for the input weights are generally mostly nonzero in value [13], providing dense connectivity, while the majority of the reservoir weights are set to zero, providing sparse connectivity. A small percentage of the reservoir weights, often as low as 5% [14] are assigned nonzero values, typically either in a random fashion or through probabilistic selection from a few preselected values. If desired, the ESN can also contain output feedback weights that are depicted with dashed lines in Fig. 1.

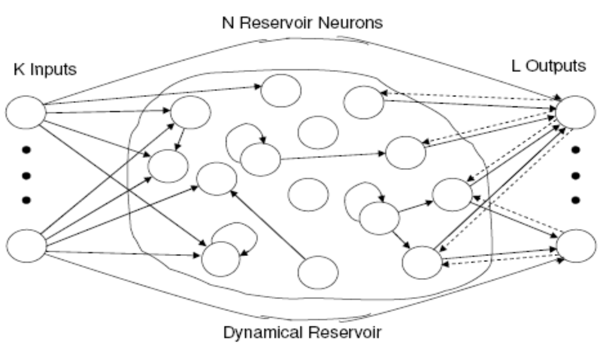

Fig. 1. Echo State Network.

The feedback weights are also not adapted during training and are instead held at fixed values that generally are mostly nonzero in value [13]. It is the fact that only the readout weights are adapted which makes the training of ESNs simpler and faster than that for conventional RNNs [11]. This adaptation for ESNs is performed with simple linear regression [13] that is discussed in section III. For a depiction of a conventional RNN along with its weights requiring adaptation, refer to [3]. For descriptions of methods used to adapt the conventional RNN weights, refer to [15] for Backpropagation Through Time, to [16] for Extended Kalman Filters, and to [3] for Particle Swarm Optimization.

The various weights are compactly stored in matrix form. For an ESN with K inputs, L outputs, and N reservoir neurons, the dimensions of the weight matrices are as follows:

- Input weights matrix $W_{in} = [N\ X\ K]$
- Reservoir weights matrix $W = [N\ X\ N]$
- Readout weights matrix $W_{out} = [L\ X\ (K + N)]$
- Optional feedback weights matrix $W_{fb} = [N\ X\ L]$

At each time step t for which inputs are fed into the ESN, the $[N\ X\ 1]$ dimension reservoir state vector $x(t+1)$ is calculated for the next time step using equation (1), where Ψ is the activation function for the reservoir neurons and $d(t)$ is the $[L\ X\ 1]$ dimension vector of target outputs that the ESN outputs are desired to approximate. The $[(N + K)\ X\ 1]$ dimension extended system state vector $z(t+1)$ is formed by concatenating $x(t+1)$ with the $[K\ X\ 1]$ dimension input vector $u(t+1)$ as shown in equation (2). The output vector $y(t+1)$ for the next time step is then calculated per equation (3), where Ψ is the activation function for the output neurons.

$$x(t+1) = \Psi(W*x(t) + W_{in}*u(t+1) + W_{fb}*d(t)) \quad (1)$$

$$z(t+1) = [x(t+1)\ ;\ u(t+1)] \quad (2)$$

$$y(t+1) = \Psi(W_{out}*z(t+1)) \quad (3)$$

In (1), $x(t)$ is set to an arbitrary value for the first time step. To work properly, the ESN must be able to asymptotically eliminate this initial condition. If it can do this, the network is said to possess the echo state property. Refer to [14] for methods of ensuring the echo state property exists for the dynamical reservoir.

III. TRAINING AND TESTING OF THE ESN

A. Training

A hyperbolic tangent activation function was used for the reservoir neurons and a linear activation function was used for the output neurons. The input weights matrix was populated using randomly selected values from a uniform distribution ranging from 0 to 1. This is also how the feedback weights matrix was populated in cases where feedback of target outputs was used. When feedback was not used, all values in this matrix were set to zero. The reservoir weights matrix was populated in a manner that guaranteed the echo state property.

The training data set consisted of 22165 input vectors spanning the time frame from 3/1/2009 through 3/31/2009 in one minute increments. Each input vector consisted of the 6 inputs listed in section I normalized to values ranging from -1 to 1. The 9 outputs calculated by the ESN were the predicted global horizontal irradiance at times of 30, 60, 90,….,270 minutes in advance of the input vector. As each input vector was fed into the ESN, the resulting *[(N + K) X 1]* dimension extended system state vector was calculated and its transpose was stored in the *[n X (N + K)]* state collection matrix *S*, where *n* is equal to the number of input vectors. Additionally, for each of the *n* time steps, the 9 target outputs were stored in the *[n X L]* dimension teacher collection matrix *D*. These target outputs were simply the measured global horizontal irradiance at times of 30, 60, 90,…,270 minutes in advance of the input vector that were retrieved from the training data set.

The asymptotic elimination of the initial condition that occurs when the echo state property is present means that it takes a number of time steps for the reservoir states to settle to converged values. Due to this phenomena, typical practice is to discard a certain number of rows *m* from the beginning of the *S* and *D* matrices [11], thus reducing their dimension to *[(n – m) X (N + K)]* and *[(n – m) X L]*, respectively.

After all the input vectors were fed into the ESN, the readout weights were adapted using linear regression. Linear regression is a technique for modeling the relationship between one or more independent variables *X* and one or more dependent variables *Y* that takes the form of (4), where ε is an error term that captures all other factors other than the independent variables that influence the dependent variables.

$$Y = X\beta + \varepsilon \tag{4}$$

For the ESN training, the teacher collection matrix *D* takes the place of *Y*, the state collection matrix *S* takes the place of *X*, the transpose of the readout weight matrix W_{out} takes the place of β, and ε is a null vector, resulting in (5).

$$D = S(Wout)^T \tag{5}$$

Some manipulations of (5) result in:

$$S^TD = S^TS(Wout)^T \tag{6}$$

$$(W_{out})^T = (S^TS)^{-1}S^TD \tag{7}$$

$$W_{out} = [(S^TS)^{-1}S^TD]^T \tag{8}$$

The training was repeated multiple times using various values for the number of reservoir neurons, the amount of connectivity within the reservoir, and the number of rows discarded from the beginning of the *S* and *D* matrices in an attempt to find the combination resulting in the best predictions. Experimentation was also done in an effort to determine how the feedback of target outputs affected the accuracy of predictions. The various values that were tried for these parameters are detailed in section IV.

B. Testing

Once the ESN was trained, it was tested using a test data set of 26629 input vectors spanning the time frame from 5/1/2009 through 5/31/2009 in one minute increments. An important change was incorporated, however, with regard to the feedback of target outputs. In real world applications, measured target values are not available at the time the ESN is calculating a predicted value for that output. To duplicate this condition, measured values from the test data set were not used as the target outputs during the testing phase even though, in this case, they were available in the test data set. This was accomplished in two different ways. The first method was simply to not incorporate feedback at all during the test phase. As detailed in section IV, this resulted in extremely poor accuracy for the test set predictions if feedback had been used to train the ESN. The second method was to originally calculate the outputs without incorporating feedback, then feed back these calculated outputs as the target values and recalculate the outputs in a recursive manner. It is expected that the more accurate the original calculated outputs were, the more accurate the recalculated outputs obtained with feedback would be. As stated previously, however, outputs obtained without feedback during testing were extremely inaccurate if the ESN had been trained using feedback. To improve accuracy, therefore, the ESN was trained initially without incorporating feedback and then outputs were obtained for the test set without incorporating feedback and stored for later use. The ESN was then re-trained using feedback of target outputs from the training data set and, finally, outputs were recalculated for the test data set using the re-trained ESN and the previously stored outputs as target values. The flowchart for this algorithm appears in Fig. 2.

IV. RESULTS

The first performance metric used to quantify the accuracy of the ESN predicted values was the linear regression correlation coefficient *r* that was calculated from a comparison of the predicted and target values as shown in Fig. 3. A second performance metric that was used was the mean squared error (MSE) between the normalized values for the predicted and target outputs. Parameters for the ESN were adjusted to maximize the average of the *r* value for the 9 ESN outputs. Once the *r* value was maximized, then the MSE was also calculated using the same ESN parameters. The only parameter that had a large effect on the correlation for the training data set was whether or not feedback of the target values was incorporated. As long as feedback was incorporated, the average value of *r* for the 9 outputs was generally always above 0.97. Table I, which reports the average results from 3 trials for each parameter setting, illustrates the relatively small effect on *r* that the number of reservoir neurons had for the training data set. Varying the ESN parameters, however, had a much larger effect on the value of *r* for the test data set. Table II, which also reports the average results from 3 trials, shows how large this effect was when, not only feedback was varied, but also when the number of reservoir neurons was varied. Due to this observation, the average *r* value of the 9 ESN outputs for the test data set is what was monitored in order to optimize the parameters.

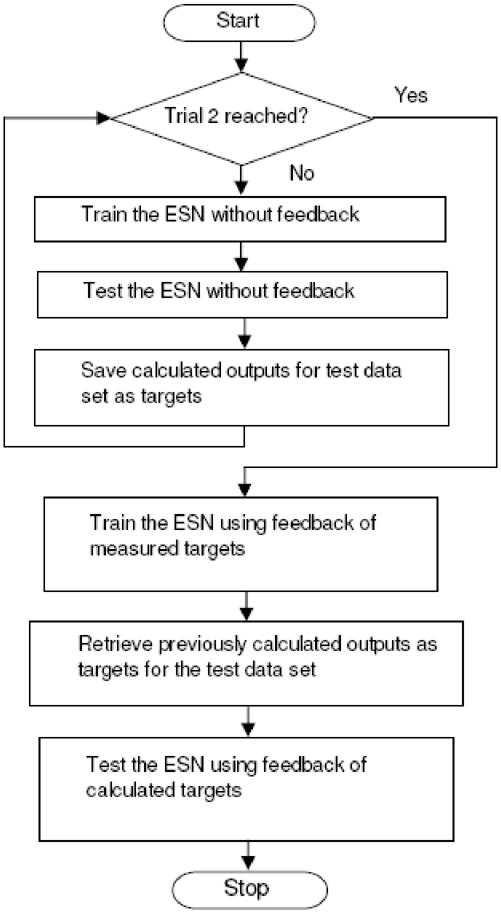

Fig. 2. Flowchart for recursive method of obtaining target outputs for the testing phase.

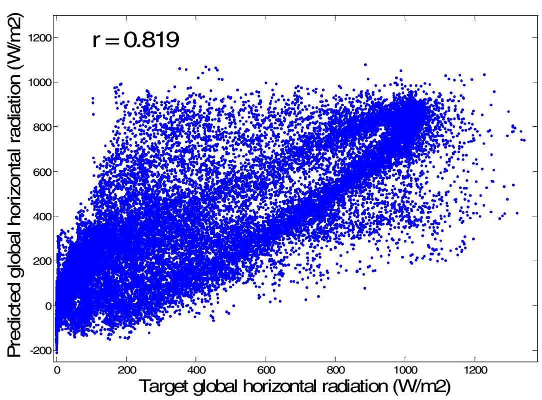

Fig. 3. Correlation between predicted and target values at $t + 60$ min.

TABLE I
EFFECT OF FEEDBACK ON TRAINING RESULTS

Reservoir neurons	Average correlation coefficient for training data set: Measured targets fed back	No feedback
100	0.983	0.695
150	0.979	0.691
200	0.971	0.674
250	0.970	0.638

TABLE II
EFFECT OF FEEDBACK ON TESTING RESULTS

	Average correlation coefficient for test data set			
Training feedback	Measured targets	Measured targets	Measured targets	None
Testing feedback	Measured targets	Calculated targets	None	None
Reservoir neurons				
100	0.954	0.187	-0.080	0.521
150	0.959	0.293	0.096	0.552
200	0.949	0.552	0.149	0.549
250	0.947	0.576	0.137	0.558

Table II indicates that r is maximized for the test data set when measured target values are fed back not only during the training phase, but also during the testing phase. Measured targets are not available when one wants to make real world predictions, however. Utilizing measured targets in the testing phase, therefore, as explained in section III, represents a condition that is not realizable and is presented for illustrative purposes only. Table II also shows the poor correlation mentioned in Section III when feedback is incorporated only during the training phase. This left two preferred methods to compare. The first method did not utilize feedback for either the testing or training data sets, while the second fed back target outputs in the testing phase that were calculated using the recursive procedure shown in Fig. 2.

It can be seen from Table II that the number of reservoir neurons greatly influenced the prediction accuracy when calculated targets were fed back, with 250 neurons being the best choice amongst those considered. With the number of neurons held constant at the optimal value of 250, the connectivity of the reservoir was then varied within the range shown in Table III, which reports the average results from 3 trials. The optimal value for this parameter was found to be 25%. The percentage of the total extended system state vectors discarded from the beginning of the state collection matrix and the percentage of the total target output vectors discarded from the beginning of the teacher collection matrix was then varied within the range shown in Table III and 7% was found to produce the best accuracy.

TABLE III
EFFECT OF PARAMETER ADJUSTMENT ON TESTING RESULTS

Training feedback	None	Measured targets	Training feedback	None	Measured targets
Testing feedback	None	Calculated targets	Testing feedback	None	Calculated targets
Reservoir connectivity	Average correlation coefficient for test data set		% of states and targets discarded during training	Average correlation coefficient for test data set	
5%	0.528	0.117	2%	0.596	0.620
10%	0.543	0.281	7%	0.593	0.621
15%	0.558	0.576	10%	0.592	0.593
20%	0.586	0.583	20%	0.580	0.578
25%	0.592	0.593			

Tables II and III indicate a relatively small change in accuracy results from changes to the ESN parameters when feedback is not utilized. The change is much larger, however, when calculated targets are fed back. If the parameters are not chosen carefully, it is clear that feeding back calculated targets will result in a lower r value than if feedback was not utilized.

A careful selection of parameters, however, resulted in the network with feedback having the higher value for this performance metric.

A total of 10 trials both with and without feedback were conducted with the ESN in the optimized configuration. Table IV shows a much higher r value for the training data set when feedback is used. This is consistent with the results in Table I. The true measure of the network's usefulness, however, is its performance on the test data set. Table IV shows the average r value for the network with feedback to be higher than that for the other network by an amount in excess of seven standard deviations. For the MSE performance metric, however, the difference is much smaller. It can also be seen that the training time with feedback is about twice of that without feedback. This is due to the fact that the network with feedback must be trained and tested twice as shown in Fig. 2.

The values of both r and MSE for each of the 9 ESN outputs for the test data set was averaged over the 10 trials and is shown in Fig. 4. The standard deviation over the 10 trials for each of these values is shown in Fig. 5. The values for MSE in both of these figures were scaled up by a factor of 10 for clearer viewing. For shorter time horizon predictions, it can be seen that the r value is approximately the same for both networks, but, for longer time horizon predictions of 180 to 270 minutes in the future, the network with feedback has a superior r value that is significant in relation to the standard deviation. For all time horizons, the difference in MSE for the predictions made with and without feedback is insignificant in relation to the standard deviation. For both networks, it can be seen that the accuracy of the predictions decreases significantly as the time horizon increases.

TABLE IV
EFFECT OF FEEDBACK ON PERFORMANCE

	Correlation coefficient for training data set				
Feedback?	Best	Worst	Average	Standard Deviation	Average MSE
no	0.590	0.578	0.583	0.003	0.0362
yes	0.960	0.955	0.958	0.002	0.0045
	Correlation coefficient for test data set				
Feedback?	Best	Worst	Average	Standard Deviation	Average MSE
no	0.599	0.589	0.594	0.004	0.0638
yes	0.628	0.617	0.622	0.003	0.0636
	Training time in seconds				
Feedback?	Best	Worst	Average	Standard Deviation	
no	39.55	50.36	43.68	3.559	
yes	83.16	85.87	84.31	0.902	

The impact of reduced accuracy as the time horizon increases is illustrated in Fig. 6 and 7. In Fig. 6, for which r = 0.874, the calculated values for global horizontal radiation at t + 30 minutes clearly approximate more closely the measured values than in Fig. 7, for which r = 0.490 at t + 270 minutes.

Implementation of the recursive feedback method depicted in Fig. 2 requires two sets of synaptic weights to be stored; one set for the ESN trained without feedback and a second set for the ESN trained while using feedback of measured targets. With this method, two calculations are made at each time step. The first calculation utilizes the weights from the ESN trained without feedback and calculates target outputs. The second calculation uses the weights from the ESN trained with feedback. For this calculation, the previously calculated targets are fed back and the ESN outputs represent the actual predictions for that time step.

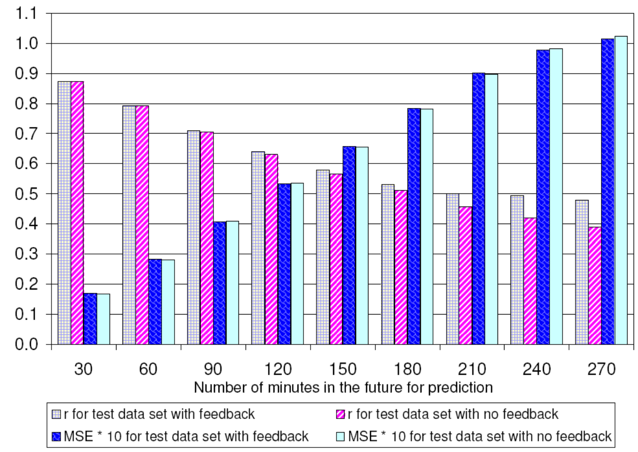

Fig. 4. Average correlation and error values for 10 trials as a function of time horizon.

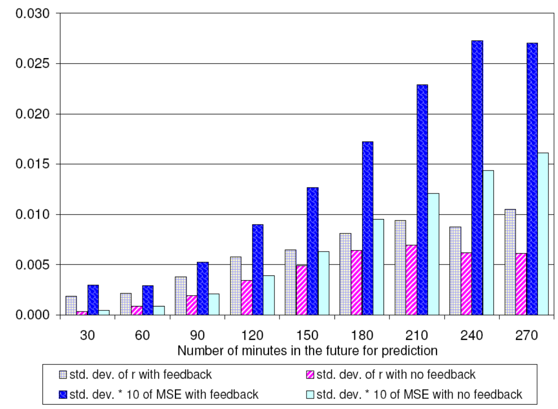

Fig. 5. Standard deviation of data from Fig. 4.

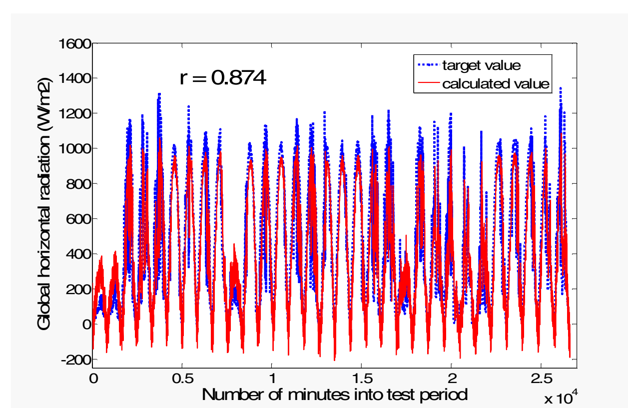

Fig. 6. Measured target and calculated values at t + 30 minutes with r = 0.874.

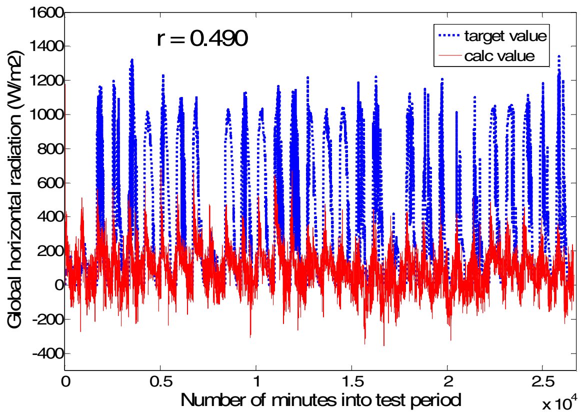

Fig. 7. Measured target and calculated values at $t + 270$ minutes with $r = 0.490$.

V. Conclusion and Future Work

The accuracy of an ESN utilizing feedback of target values is limited by the fact that, in real world applications, the only available targets are imperfect ones obtained from calculations done without the benefit of feedback. Nonetheless, this technique has been demonstrated to produce targets of sufficient accuracy to improve the correlation coefficient of solar irradiance predictions for time horizons in the range of 180 to 270 minutes. A similar improvement with regard to the mean squared error performance metric has not been demonstrated. Regardless of whether or not feedback is incorporated, the accuracy of predictions decreases significantly as the time horizon increases. For this study, the correlation coefficient steadily decreased from approximately 0.87 on average for predictions made at $t + 30$ minutes to approximately 0.48 on average for predictions made at $t + 270$ minutes when feedback was used.

Future work exists in evaluating why different results were obtained with the two performance metrics. Performance metrics should also be obtained for the case where the ESN is both trained and tested for the same month from different years instead of the different months of March and May. Additional future work involves determining to what extent accuracy would be affected by removing global horizontal irradiance at time t from the ESN inputs. If global horizontal irradiance could be removed as an input, then this ESN could be applied in locations where instrumentation for measuring this value is not available.

References

[1] Energy Information Administration (EIA) Renewable Energy Consumption and Electricity Preliminary 2007 Statistics (online): http://www.eia.doe.gov/cneaf/alternate/page/renew_energy_consump/rea_prereport.html.

[2] Energy Information Administration (EIA) Solar Photovoltaic Cell/Module Manufacturing Activities (online): http://www.eia.doe.gov/cneaf/solar.renewables/page/solarphotv/solarpv.html.

[3] Welch RL, Ruffing SM, Venayagamoorthy GK, "Comparison of Feedforward and Feedback Neural Network Architectures for Short Term Wind Speed Prediction", *Proceedings, International Joint Conference on Neural Networks*, June 2009, pp. 3335-3340.

[4] Fadare DA, "Modelling of Solar Energy in Nigeria Using an Artificaial Neural Network Model", Elsevier Applied Energy Journal, vol. 86, issue 9, September 2009, pp. 1410-1422.

[5] Senkal O, Kuleli T, "Estimation of Solar Radiation Over Turkey Using Artificial Neural Network and Satellite Data", Elsevier Applied Energy Journal, vol. 86, issues 7-8, July-August 2009, pp. 1222-1228.

[6] Benghanem M, Mellit A, Alamri SN, "ANN-based Modelling and Estimation of Daily Global Solar Radiation Data: A Case Study" Elsevier Energy Conservation and Management Journal, vol. 50, issue 7, July 2009, pp. 1644-1655.

[7] Azadeh A, Maghsoudi A, Sohrabkhani S, "An Integrated Artificial Neural Networks Approach for Predicting Global Radiation", Elsevier Energy Conservation and Management Journal, vol. 50, issue 6, June 2009, pp. 1497-1505.

[8] Lorenz E, Hurka J, Heinemann D, Beyer HG, "Irradiance Forecasting for the Power Prediction of Grid-Connected Photovoltaic Systems" *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 2, no. 1, March 2009, pp. 2-10.

[9] Tindal A, Parkes J, Munoz L, Wasey J, "Wind Energy Trading Benefits Through Short Term Forecasting", Proceedings of the European Wind Energy Conference, February – March 2006, BB1.5.

[10] Moreno-Munoz A, de la Rosa JJG, Posadillo R, Bellido F, "Very Short Term Forecasting of Solar Radiation", *33rd IEEE Photovoltaic Specialists Conference, May 2008,* pp. 1-5.

[11] Jaeger, H, "Echo State Network", Scholarpedia article (online): http://www.scholarpedia.org/article/Echo_state_network.

[12] National Renewable Energy Laboratory's Solar Radiation Research Laboratory (online): http://www.nrel.gov/midc/srrl_bms/.

[13] Lukosevicius M, Jaeger H, "Reservoir Computing Approaches to Neural Network Training" Computer Science Review, March 2009.

[14] Jaeger, H, "The Echo State Approach to Analysing and Training Recurrent Neural Networks", Tech. Report No. 148, German National Research Center for Information Technology, 2001.

[15] Werbos PJ, "Backpropagation Through Time: What It Is and How to Do It", Proceedings of the IEEE, vol. 78, issue 10, October 1990, pp. 1550-1560.

[16] Puskorius GV, Feldkamp LA, "Neurocontrol of Nonlinear Dynamical Systems with Kalman Filter Trained Recurrent Networks", IEEE Transactions on Neural Networks, vol. 5, issue 2, March 1994, pp. 279-297.