



HAL
open science

Uncertainty modeling in wind power generation prediction by neural networks and bootstrapping

Ronay Ak, Valeria Vitelli, Enrico Zio

► **To cite this version:**

Ronay Ak, Valeria Vitelli, Enrico Zio. Uncertainty modeling in wind power generation prediction by neural networks and bootstrapping. ESREL 2013, Sep 2013, Amsterdam, Netherlands. pp.1-6. hal-00838772

HAL Id: hal-00838772

<https://hal-centralesupelec.archives-ouvertes.fr/hal-00838772>

Submitted on 26 Jun 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Uncertainty modeling in wind power generation prediction by neural networks and bootstrapping

R. Ak & V. Vitelli

Chair of Systems Science and the Energetic Challenge, European Foundation for New Energy - Électricité de France, École Centrale Paris and Supélec, France.

E. Zio

*Chair of Systems Science and the Energetic Challenge, European Foundation for New Energy - Électricité de France, École Centrale Paris and Supélec, France.
Energy Department, Politecnico di Milano, Italy.*

ABSTRACT: Accurate short-term wind power forecasting with quantification of the associated uncertainty is crucial for the management of energy systems including wind power generation. On top of the inherent uncertainty in wind speed, it is necessary to account also for the uncertainty in the relationship between wind speed and the corresponding power production, typically described by a power curve whose characteristic parameters are not precisely known in practice. In this paper, we propose a novel approach to wind power forecasting with uncertainty quantification. The approach can be schematized in two steps: first, short-term estimation of wind speed prediction intervals (PIs) is performed within a multi-objective optimization framework worked out by non-dominated sorting genetic algorithm-II (NSGA-II); then, the uncertainty in wind speed and the uncertainty in the power curve are combined via a bootstrap sampling technique, thus obtaining wind power PIs with same coverage as the wind speed PIs.

1 INTRODUCTION

Power production via renewable energy sources is a hot topic of research and application. This is due to both the widespread availability of such sources (e.g. wind, sun, etc.) and to the sustainability of the associated production process. Among renewable energy sources, wind power is widely recognized as one of the most promising, because of its tremendous potential in commercialization and bulk power generation.

The management of wind power generation systems relies on short-term wind power generation forecasting, which must also provide a measure of the associated uncertainty. Two uncertainty sources can be considered: the inherent uncertainty in wind speed, due to the intermittent and unstable nature of wind (aleatory uncertainty); the uncertainty in the relationship between wind power and wind speed (epistemic uncertainty) (Helton 1994). The latter uncertainty is mainly due to the parameters defining the power curve (cut-in, rated and cut-off speeds, and rated power), which can be different for each single turbine within a wind farm (Novoa & Jin 2011).

In the present work, we treat the power curve parameters as random variables and account for the epistemic uncertainty by bootstrapping (Efron 1981), which allows combining also the aleatory uncertainty in the wind speed.

To do so, we first perform short-term forecasting of wind speed in a multi-objective optimization framework, where the non-dominated sorting genetic algorithm-II (NSGA-II) (Deb et al., 2002) is applied to optimize the weights of a neural network (NN) for estimating the prediction intervals (PIs) of wind speed. We, then, combine the uncertainty in wind speed forecasting with the uncertainty in the power curve via a bootstrap sampling technique. This results in obtaining wind power PIs with the associated uncertainty. By a precise probabilistic formulation, we show that the coverage probability of the wind power PIs obtained is the same as the one of wind speed PIs. Moreover, we test the robustness of the procedure with respect to the form of the distributions for the power curve random parameters.

The rest of the paper is organized as follows. In Section 2, the methodology for NN-based wind speed PIs estimation and for bootstrap-based wind power PIs estimation is introduced and described. In Section 3, a case study is carried out to test the effectiveness of the proposed approach. Finally, in the Conclusion Section some final remarks are given.

2 METHODOLOGY

2.1 Estimation of Wind Speed PIs by NSGA-II

A PI is comprised of upper and lower bounds in which a future unknown value of the target is expected to lie with a predetermined confidence level $(1-\alpha)$. The formal definition of a PI is thus (Geisser, 1993):

$$P(L(x) < y(x) < U(x)) = 1-\alpha \quad (1)$$

where $L(x)$ and $U(x)$ indicate respectively the lower and upper bounds of the PI of the output $y(x)$ corresponding to input x ; the confidence level $(1-\alpha)$ refers to the expected probability that the true value of $y(x)$ lies within the PI, $[L(x), U(x)]$.

In order to provide wind speed PIs, we use multi perceptron artificial neural networks (NNs) (Korbicz et al. 2004) which are a class of nonlinear statistical models inspired by brain architecture, capable of learning complex nonlinear relationships among variables from observed data (Hornik et al. 1989), by a process of parameter tuning called “training”. It is common to represent the task of such a NN model as one of nonlinear regression of the kind (Zio 2006, Shrestha & Solomatine 2006):

$$y(x) = f(x; w) + \varepsilon(x), \quad \varepsilon(x) \sim N(0, \sigma_\varepsilon^2(x)) \quad (2)$$

where x , $y(x)$ are the input and output vectors of the regression, which in our case represent measured historical wind speeds at time $t-1, t-2, \dots, t-r$ and the true target at time t , respectively. w represents the vector of values of the parameters of the model function f , in general nonlinear. The term $\varepsilon(x)$ is the error associated to the regression model f , and it is assumed normally distributed with zero mean.

We evaluate the PIs by the coverage probability of the prediction intervals (CP), which one wants to maximize, and the interval width (PIW), which one wants to minimize. The mathematical definitions of the PICP and PIW used in this work are (Khosravi et al. 2011):

$$PICP = \frac{1}{n_p} \sum_{i=1}^{n_p} c_i \quad (3)$$

where n_p is the number of samples in the training or testing sets, and $c_i = 1$, if $y_i \in [L(x_i), U(x_i)]$ and $c_i = 0$ otherwise. $L(x_i)$ and $U(x_i)$ are the estimated lower and upper bounds of the prediction interval in output, in correspondence of the input x_i .

$$NMPIW = \frac{1}{n_p} \sum_{i=1}^{n_p} \frac{(U(x_i) - L(x_i))}{t_{max} - t_{min}} \quad (4)$$

where NMPIW is the Normalized Mean PIW, and t_{min} and t_{max} represent the true minimum and

maximum values of the targets (i.e., the bounds of the range in which the true values fall) in the training set, respectively. Normalization of the PI width by the range of targets makes it possible to objectively compare the PIs, regardless of the techniques used for their estimation or the magnitudes of the true targets.

The PIs estimation problem is addressed by taking into account the two conflicting objectives in a multi-objective framework. For this, we use NSGA-II, which is one of the most efficient multi-objective genetic algorithms (MOGAs) (Konak et al. 2006, Deb et al. 2002), to optimize the parameters (i.e. the weights) of the network taking into account both objectives. More precisely, the neural network is trained by NSGA-II to produce the lower and upper bounds of the prediction intervals for short-term forecasting (1-hour ahead) of wind speed. The practical implementation of NSGA-II on our specific problem involves two phases: initialization and evolution. These can be summarized as follows:

Initialization phase:

Step 1: Split the input data into training (D_{train}) and testing (D_{test}) subsets.

Step 2: Fix the maximum number of generations and the number of chromosomes (individuals) Nc in each population; each chromosome codes a solution by G real-valued genes, where G is the total number of parameters (weights) in the NN. Set the generation number $n = 1$. Initialize the first population P_n of size Nc , by randomly generating Nc chromosomes.

Step 3: For each input vector x in the training set, compute the lower and upper bound outputs of the Nc NNs, each one with G parameters.

Step 4: Evaluate the two objectives PICP and NMPIW for the Nc NNs (one pair of values 1-PICP and NMPIW for each of the Nc chromosomes in the population P_n).

Step 5: Rank the chromosomes (vectors of G values) in the population P_n by running the fast non-dominated sorting algorithm (Deb et al. 2002) with respect to the pairs of objective values, and identify the ranked non-dominated fronts F_1, F_2, \dots, F_k where F_1 is the best front, F_2 is the second best front and F_k is the least good front.

Step 6: Apply to P_n a binary tournament selection based on the crowding distance (Deb et al. 2002), for generating an intermediate population S_n of size Nc .

Step 7: Apply the crossover and mutation operators to S_n , to create the offspring population Q_n of size Nc .

Step 8: Apply Step 3 onto Q_n and obtain the lower and upper bound outputs.

Step 9: Evaluate the two objectives in correspondence of the solutions in Q_n , as in Step 4.

Evolution phase:

Step 10: If the maximum number of generations is reached, stop and return P_n . Select the first Pareto front F_1 as the optimal solution set. Otherwise, go to Step 11.

Step 11: Combine P_n and Q_n to obtain a union population $R_n = P_n \cup Q_n$.

Step 12: Apply Steps 3-5 onto R_n and obtain a sorted union population.

Step 13: Select the N_c best solutions from the sorted union to create the next parent population P_{n+1} .

Step 14: Apply Steps 6-9 onto P_{n+1} to obtain Q_{n+1} . Set $n = n + 1$; and go to Step 10.

Finally, the best front in terms of ranking of non-dominance and diversity of the individual solutions is chosen. Once the best front of solutions is obtained, then the testing step is performed on the trained NN with optimal weight values.

2.2 Wind Power PIs Estimation

The wind power value $p(x)$ depends on the wind speed $y(x)$. Suppose that $[L_p(x), U_p(x)]$ is the PI associated to the wind power value $p(x)$ in correspondence of the input x , i.e. to the wind speed value $y(x)$. Then, the following property must hold:

$$P(L_p(x) \leq p(x) \leq U_p(x)) = 1 - \alpha_p, \quad (5)$$

where $1 - \alpha_p \in [0,1]$ is the coverage probability.

Our working hypothesis stands on the fact that both wind power values and PIs depend on the wind speed values and PIs, respectively, via a non-monotonic transformation, namely the power curve. In this hypothesis, the rest of the subsection is devoted to the following two issues:

1. assess the value of $1 - \alpha_p$ given the coverage probability of the PI associated to the wind speed $y(x)$;
2. develop a bootstrap-based approach to the estimation of $[L_p(x), U_p(x)]$.

In order to assess the coverage probability of wind power PIs, we have to take into account the fact that they have been obtained via a power curve transformation, which means:

$$L_p(x) = g_\theta(L(x)) \quad (6)$$

$$U_p(x) = g_\theta(U(x)) \quad (7)$$

where $[L(x), U(x)]$ is the PI for the wind speed value $y(x)$ associated to the input x , with associated coverage probability $1 - \alpha_s$, while g_θ is a quadratic power curve transformation given by the following expression (Justus et al. 1976):

$$g_\theta(V) = \begin{cases} 0 & \text{if } V \leq V_{ci} \text{ or } V > V_{co} \\ P_r(a + bV + cV^2) & \text{if } V_{ci} < V < V_r \\ P_r & \text{if } V_r < V \leq V_{co} \end{cases} \quad (8)$$

with

$$a = -\frac{V_{ci}(V_{ci}+V_r).(V_{ci}^2+2V_{ci}V_r-V_r^2)}{2(V_{ci}-V_r)^2.V_r^2} \quad (9)$$

$$b = \frac{V_{ci}^4+4V_{ci}^3.V_r+6V_{ci}^2.V_r^2-2V_{ci}V_r^3-V_r^4}{2(V_{ci}-V_r)^2.V_r^3} \quad (10)$$

$$c = -\frac{V_{ci}^3+3V_{ci}^2.V_r+3V_{ci}V_r^2-3V_r^3}{2(V_{ci}-V_r)^2.V_r^3} \quad (11)$$

and with $\theta = (V_{ci}, V_r, V_{co}, P_r)$ being the vector of parameters defining the power curve, i.e. cut-in speed, rated speed, cut-off speed and rated power. A plot of the power curve g_θ is shown in Figure 1. In the following, we will consider V_{co} and P_r to be fixed (deterministic) values, and respectively equal to the values 30 m/s and 20 MW (Albadi & El-Saadany 2012, Akdag & Guler 2010), while V_{ci} and V_r are random variables with distributions F_{ci} and F_r , respectively. The inherent stochasticity in the power curve is motivated by the fact that different wind turbines correspond to specific power curve parameters, which leads to an imprecise and imperfect knowledge of the power curve transformation.

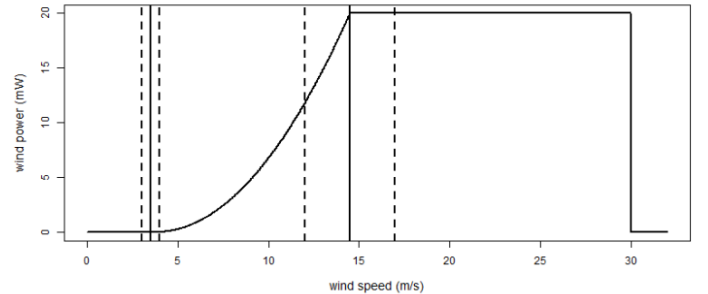


Figure 1. Plot of the power curve g_θ as a function of wind speed. Solid vertical lines correspond to the values of the two stochastic parameters V_{ci} and V_r . Dashed vertical lines identify the domains of the distributions F_{ci} and F_r , respectively.

The following chain of identities holds:

$$P(L_p(x) \leq p(x) \leq U_p(x)) = \quad (12)$$

$$= P(g_\theta(L(x)) \leq p(x) \leq g_\theta(U(x))) =$$

$$= \int_{\theta \in \Theta} P(g_\theta(L(x)) \leq p(x) \leq g_\theta(U(x)) | \theta = \theta_0) \cdot P(\theta = \theta_0) \cdot d\theta =$$

$$= \int_{\theta \in \Theta} P(g_\theta^{-1}(g_\theta(L(x))) \leq y(x) \leq g_\theta^{-1}(g_\theta(U(x))) | \theta = \theta_0) \cdot P(\theta = \theta_0) \cdot d\theta =$$

$$= \int_{\theta \in \theta} (1 - \alpha_s) P(\theta = \theta_0) d\theta = (1 - \alpha_s).$$

The first and third equalities in Equation (12) derive from our working hypothesis, the second one from the theorem of total probability, the fourth one from the definition of coverage probability for wind speed PIs, and the last one stands because we integrate out the parameter vector θ over the whole probability space θ .

Hence, we can conclude that $\alpha_p = \alpha_s$, i.e. the coverage probability is maintained while passing from wind speed PIs to wind power PIs via a wind power curve transformation.

We remark that, in general, $P(f(L) \leq f(x) \leq f(U)) = P(L \leq x \leq U)$ if and only if f is a strictly monotonic function, because in this case the existence of the inverse is ensured. The power curve transformation g_θ , whose definition is given in Equation (8), is non-monotonic, but it is monotonic when restricted to the open subset of the co-domain $(0, P_r)$. Note that the co-domain of the power curve is given by the closure of the latter subset, i.e. $[0, P_r]$. Hence, we can restrict our analysis to the open subset $(0, P_r)$, and treat the non-monotonicity issue as a border issue, a posteriori restricting the obtained wind power PIs to their domain of admissibility (note that this is usually done in the context of PIs estimation when the target of interest is a bounded variable, e.g. a proportion).

We now move to the problem of estimating the wind power PIs $[L_p(x_1), U_p(x_1)], \dots, [L_p(x_n), U_p(x_n)]$ corresponding to the testing set $\{x_i, y_i\}$, for $i = 1, \dots, n$.

Since the parameter vector θ is a multivariate random variable, the wind power PIs estimation process provides a distribution of intervals accounting for the parameters stochasticity. To get such a distribution, parametric bootstrap (Efron 1981, Shao & Tu 1995) is used. Parametric bootstrap is a technique which allows generating a sample for each parameter, and then estimating some relevant quantities concerning the target of interest.

More precisely, given the estimated wind speed PIs $[L(x_1), U(x_1)], \dots, [L(x_n), U(x_n)]$ in the testing set, the parametric bootstrap sampling technique is articulated in the following two steps:

1. Bootstrap Phase:

Sample two values for the stochastic parameters V_{ci} and V_r from the corresponding distributions, i.e. $V_{ci} \sim F_{ci}$ and $V_r \sim F_r$, and transform all wind speed PIs $[L(x_1), U(x_1)], \dots, [L(x_n), U(x_n)]$ into wind power PIs $[L_p(x_1), U_p(x_1)], \dots, [L_p(x_n), U_p(x_n)]$ via the power curve transformation g_θ defined in (8), and using the previously sampled parameter values. Repeat the sampling until a

sufficient number of sets of wind power PIs has been obtained.

2. Aggregation Phase:

Aggregate the results of the bootstrap phase by computing, for each element of the testing set, the bootstrapped average wind power PI and the 5th and 95th percentiles of the wind power PI bootstrapped distribution.

This bootstrapping technique allows obtaining a set of wind power PIs, which accounts for both the aleatory uncertainty intrinsic in wind power generation and the variability associated to wind power PIs themselves, thus expressing the epistemic uncertainty related to the power curve estimation procedure. The aleatory uncertainty can be expressed by showing the bootstrapped average wind power PIs, while the epistemic one can be summarized in the 5th and 95th percentiles of the wind power PIs bootstrapped distribution.

3 CASE STUDY

The wind speed data used in this study have been measured for Regina, Saskatchewan, a region of central Canada (Canadian Weather Office, 2012) over a period of two months from 1st of February 2012 to 31st of March 2012. The total data set includes 1437 samples (see Fig. 2), among which the first 80% (the first 1150 samples) are used for training and the rest for testing. The architecture of the multi-perceptron NN consists of one input, one hidden and one output layers. The number of input neurons is 3 corresponding to the wind speed values of the previous three time steps (W_{t-1} , W_{t-2} and W_{t-3}); the number of hidden neurons is set to 10 after a trial-and-error process; the number of output neurons is 2, one for the lower and one for the upper bound values of the PIs. As activation functions, the hyperbolic tangent function is used in the hidden layer and the logarithmic sigmoid function is used at the output layer (these choices have been found to give the best results by trial and error, although the results have not shown a strong sensitivity to them). The training of the NN weights is done by NSGA-II to maximize PICP (Equation 3) while minimizing (Equation 4). All data have been normalized within the range [0.1, 0.9]. To account for the inherent randomness of NSGA-II, twenty different runs have been performed and an overall best non-dominated Pareto front has been obtained from the twenty individual fronts.

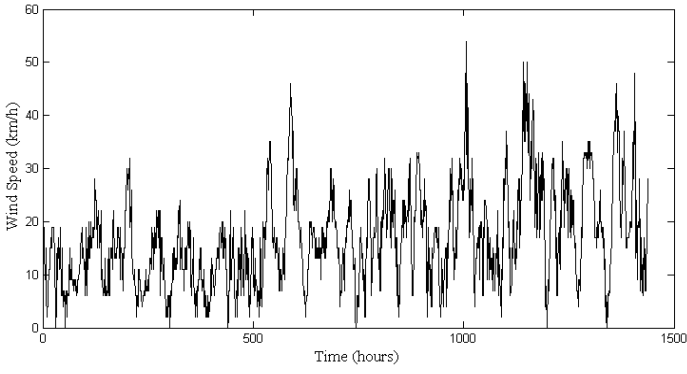


Figure 2. The wind speed data set used in this study.

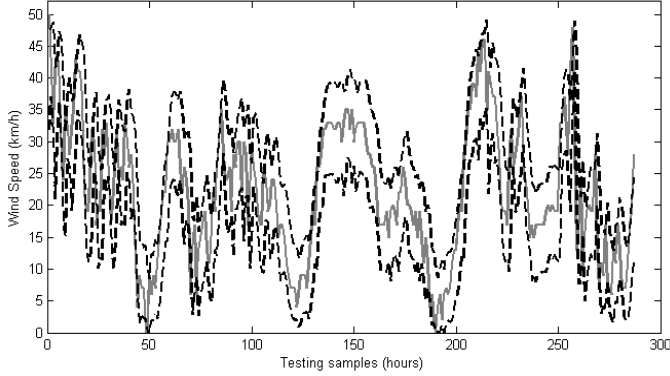


Figure 3. Estimated PIs for 1-hour ahead wind speed prediction on the test data set (dashed lines), and wind speed target data included in the test data set (solid line).

Given the overall best Pareto set of optimal solutions (i.e. optimal NN weights), one has to pick one (i.e. one trained NN) for use. We take a solution subjectively chosen, because judged to provide a good compromise in terms of high PICP and low NMPIW. The selected solution has 90% CP and 0.242 NMPIW on the training, and 82% CP and 0.255 NMPIW on the testing. Figure 3 shows wind speed target data (testing set) together with the estimated PIs corresponding to the selected solution.

The bootstrapping estimation technique described in the previous section is then applied to the estimated wind speed PIs (testing set) shown in Figure 3 to obtain wind power PIs. The number of bootstrap replicates has been set equal to 1000. In order to test the robustness of this bootstrapping technique with respect to the parametric assumption concerning the distribution of the power curve parameters, we sample V_{ci} and V_r from both a uniform and a Gaussian distributions centered around average values of 3.5 and 14.5 m/s, respectively, with a range of uncertainty of [3, 4] and [12, 17] m/s, respectively, defining the domain of the associated distribution (see Figure 1). Then, the two parameters are sampled either from a uniform distribution ($F_{ci} = U[3,4]$ and $F_r = U[12,17]$), or from a Gaussian one ($F_{ci} = N(3.5, (1/6)^2)$ and $F_r = N(14.5, (5/6)^2)$).

The resulting average bootstrapped PIs for 1-hour ahead wind power prediction, obtained by applying to the wind speed PIs of the testing data set the bootstrapping scheme described in the previous section, are shown in Figure 4. From inspection, the robustness of the bootstrapping procedure with respect to the distribution hypothesis can be appreciated. The results are also compared with the ones obtained by fixing the stochastic parameters defining the power curve to their average values; in this case, the uncertainty is evidently underestimated.

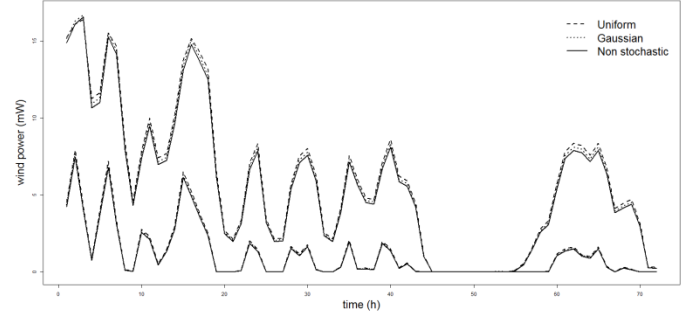


Figure 4. Average bootstrapped PIs for 1-hour ahead wind power prediction on the testing data set, obtained by sampling the power curve stochastic parameters from a uniform distribution (dashed lines), from a Gaussian distribution (dotted lines), and by fixing them to their average values (solid lines).

In Figures 5 and 6, we finally show the bootstrapped distributions of the wind power PIs obtained by uniform and Gaussian sampling, respectively. The bootstrapped distributions are shown by the 5th and 95th percentiles (dotted and dashed lines, respectively). By looking at these plots, some considerations can be made: first, the bootstrapping technique allows us to efficiently decouple epistemic (PIs distribution) and aleatory (PIs width) uncertainty. Secondly, the epistemic uncertainty that generates a variability into the PIs bounds, described by the percentiles in Figures 5 and 6, is smaller than the aleatory uncertainty, quantified via the PIs width: this can be appreciated in the fact that the 95th percentile of the lower bound bootstrapped distribution is never greater than the 5th percentile of the upper bound bootstrapped distribution; or, in other words, by the fact that the PIs width is in general bigger than the uncertainty associated to the PIs themselves.

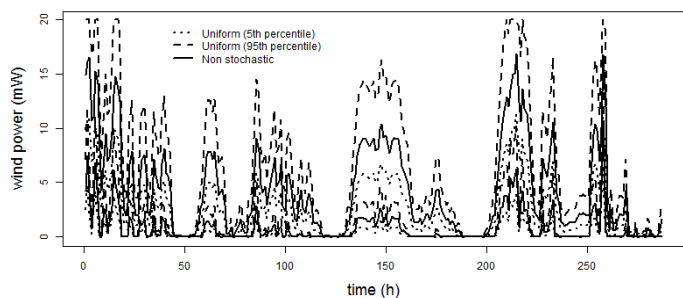


Figure 5. 5th (dotted lines) and 95th (dashed lines) percentiles of the bootstrapped distribution of 1-hour ahead wind power PIs obtained by sampling the power curve stochastic parameters from a uniform distribution, together with wind power PIs obtained by fixing the parameters to their average values (solid line).

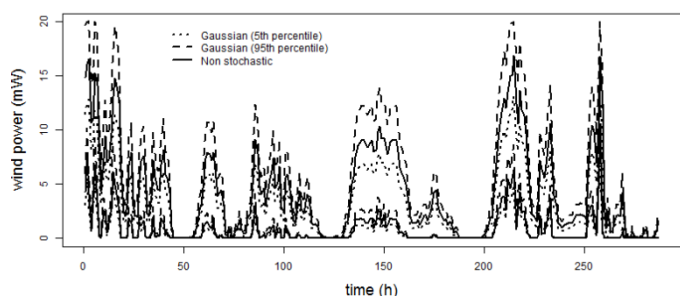


Figure 6. 5th (dotted lines) and 95th (dashed lines) percentiles of the bootstrapped distribution of 1-hour ahead wind power PIs obtained by sampling the power curve stochastic parameters from a Gaussian distribution, together with wind power PIs obtained by fixing the parameters to their average values (solid line).

4 CONCLUSIONS

In this work, we presented a novel approach to wind power PIs estimation, taking into account both aleatory and epistemic uncertainty. The proposed approach quantifies aleatory uncertainty by estimating wind speed PIs, and then transforms them into wind power PIs by using a power curve. In doing so, epistemic uncertainty arising from the imperfect knowledge of the power curve parameters is also taken into account through bootstrap sampling. The procedure manages to effectively decouple aleatory and epistemic uncertainty, and moreover shows a good robustness with respect to the parametric assumptions implicit in the bootstrap. The invariance of the coverage probability by passing from wind speed to wind power PIs has also been shown.

REFERENCES

Akdag, A. A. & Güler, Ö. 2011. Comparison of wind turbine power curve models. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects* 33 (24): 2257-2263.

- Albadi, M. & El-Saadany, E. 2012. Comparative study on impacts of power curve model on capacity factor estimation of pitch-regulated turbines. *TJER* 9: 36–45.
- Canadian Weather Office. 2012. Website: http://www.weatheroffice.gc.ca/canada_e.html, (accessed Dec., 2012).
- Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T. 2002. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6 (2): 182–197.
- Efron, B. 1981. Nonparametric estimates of standard error: the jackknife, the bootstrap and other methods. *Biometrika* 68 (3): 589–599.
- Geisser, S. 1993. *Predictive Inference: An Introduction*. Chapman and Hall.
- Helton, J.C., 1994. Treatment of uncertainty in performance assessments for complex systems. *Risk Analysis* 14 (4): 483–511.
- Hornik, K., Stinchcombe, M. & White, H. 1989. Multilayer feedforward networks are universal approximators. *Neural Networks* 2: 359–366.
- Justus, C.G., Hargraves, W.R. & Yalcin, A. 1976. Nationwide assessment of potential output from wind-powered generators. *Journal of Applied Meteorology* 15: 673–678.
- Khosravi, A., Nahavandi, S., Creighton, D. & Atiya, A.F. 2011. Lower upper bound estimation method for construction of neural network-based prediction intervals. *IEEE Transactions on Neural Networks* 22 (3): 337–346.
- Konak, A., Coit, D.W. & Smith, A.E. 2006. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability Engineering & System Safety* 91(9): 992–1007.
- Korbicz, J., Koscielny, J.M., Kowalczyk, Z. & Cholewa, W. 2004. *Fault diagnosis: models, artificial intelligence, applications*. Berlin: Springer-Verlag.
- Novoa, C. & Jin, T. 2011. Reliability centered planning for distributed generation considering wind power volatility. *Electric Power Systems Research* 81(8): 1654–1661.
- Shao, J. & Tu, D. 1995. *The Jackknife and Bootstrap*. Springer Series in Statistics, New York.
- Shrestha, D.L. & Solomatine, D.P. 2006. Machine learning approaches for estimation of prediction interval for the model output. *Neural Networks* 19: 225–235.
- Zio, E. 2006. A study of the bootstrap method for estimating the accuracy of artificial neural networks in predicting nuclear transient processes. *IEEE Transactions on Nuclear Science* 53 (3): 1460–1478.