

Predicting Ramp Events with a Stream-based HMM framework

Carlos A. Ferreira¹ and João Gama² and Vítor S. Costa³ and Vladimiro Miranda⁴ and Audun Botterud⁵

Abstract. The motivation for this work is the study and prediction of wind ramp events occurring in a large-scale wind farm located in the US Midwest. In this paper we introduce the SHREA framework, a stream-based model that continuously learns a discrete HMM model from wind power and wind speed measurements. We use a supervised learning algorithm to learn HMM parameters from discretized data, where ramp events are HMM states and discretized wind speed data are HMM observations. The discretization of the historical data is obtained by running the SAX algorithm over the first order variations in the original signal. SHREA updates the HMM using the most recent historical data and includes a forgetting mechanism to model natural time dependence in wind patterns. To forecast ramp events we use recent wind speed forecasts and the Viterbi algorithm, that incrementally finds the most probable ramp event to occur.

We compare SHREA framework against Persistence baseline in predicting ramp events occurring in very short-time horizons.

1 Introduction

Ramping is one notable characteristic in a time series associated with a drastic change in value in a set of consecutive time steps. Two properties of a ramping event i.e. slope and phase error, are important from the point of view of the System Operator (SO), with important implications in the decisions associated with unit commitment or generation scheduling. Unit commitment decisions must prepare the generation schedule in order to smoothly accommodate forecasted drastic changes in wind power availability [2]. In this paper we present SHREA a novel stream-based framework that predicts ramping events in short term wind power forecasting.

The development of the SHREA framework is the answer to the three main issues available in ramp event forecasting. How can we describe and get insights on the wind power, and wind speed, time-dependent dynamic and use this description to predict short-time ahead ramp events? How can we combine real valued historical wind power and speed measurements and Numerical Weather Predictions (NWP), specially wind speed predictions, to output reliable real-time predictions? How can we continuously adapt SHREA to accommodate different natural weather regimes yet producing reliable predictions?

To answer these questions we designed a stream-based framework that continuously learns a discrete Hidden Markov Model (HMM) and uses it to generate predictions. To learn and update the HMM the SHREA framework uses a supervised strategy whereas the HMM

parameters are estimated from historical data, the state transitions probabilities are estimated from wind power measurements and the emission probabilities, at each state, are estimated from wind speed observations. To estimate the state probability transitions, first, we combine a ramp filter, a derivative alike filter, and a user-defined threshold to translate the real-valued wind power time series into a labeled time-series, coding three different types of ramp events: ramp-up, no-ramp and ramp-down. Then, the transitions occurring in this labeled time series are used to estimate the transitions of the Markov process hidden in the HMM, i.e., to model the transitions between the three states associated with the three types of ramp events. To learn the HMM emission probabilities, first we combine a ramp filter and the SAX algorithm [9] to translate the wind speed measurements signal into a string. Next we use both the wind power labeled time series and the wind speed string to estimate the emission probabilities at each state. The estimative is obtained by counting the string symbols, coding wind speed variations, associated with a given state/ramp event.

When we analyze wind power historical data we observe both seasonal weather regimes and short-time ahead dependence of the recent past wind power/speed measurements. Thus, to accommodate these issues, in SHREA we included a strategy that forgets old weather regimes and continuously updates the HMM with the most recent measurements, both wind power measurements and wind speed measurements.

To generate ramp event predictions occurring in short-time ahead window we use the wind speed forecast, obtained from a major NWP provider, and the current HMM. First, we run a filter over the wind speed forecast signal to obtain a signal of wind speed variations. Next, we run the SAX algorithm to translate the resulting real-valued time series into a string. Then, we run the Viterbi algorithm [12] to obtain the most likely sequence of ramp events. We could use the Forward-Backward algorithm [12] usually used to estimate the posterior probability but we would be using long time ahead, thus unreliable, wind speed forecasts to predict current ramp events.

It is important to observe that wind speed measurements and forecasts, mainly short time horizon predictions, are approximately equally distributed over time. Moreover, the wind power output of each turbine is related to wind speed measurements.

In this work we run the SHREA framework to describe and predict very short-time ahead ramp events occurring in a large-scale wind farm located in the US Midwest. We present a comparison against the Persistence model that is known to be hard to beat in short-time forecasts [10].

Despite the difficulty of the ramp forecasting problem, in this work we make the following contributions: Develop a stream-based framework that predicts ramp events and generates both descriptive and

¹ LIAAD-INESC TEC and ISEP - Polytechnic Institute of Porto, Portugal

² LIAAD-INESC TEC and FEP - University of Porto, Portugal

³ CRACS-INESC TEC and FC - University of Porto, Portugal

⁴ INESC TEC and FE - University of Porto, Portugal

⁵ Argonne National Laboratory, Argonne, IL, USA

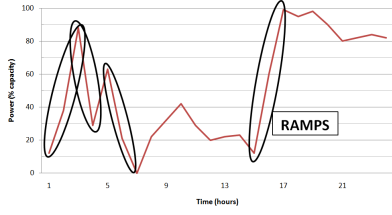


Figure 1: Illustration of ramp events, defined as a change of at least 50% in power in an interval of 4 hours

cost-effective models; Introduce a forgetting mechanism so that we can learn a HMM using only the most recent weather regimes; Use wind speed forecasts as observations of a discrete HMM to predict short-time ahead ramp events.

In the next Section we introduce the ramp event forecast problem. In Section 3 we present a detailed description of our framework. In Section 4 we present and discuss the obtained results. Last, we present some conclusions and present future research directions.

2 Ramp Event Definition and Related Work

One of the main problems in ramp forecasting is how to define a ramp. In fact, there is no standard definition [7, 3, 8] and almost all existing literature report different definitions, depending, for instance, on the location or on the farm’s size.

The authors in [5] and [11] define several relevant characteristics for ramp definition, characterization and identification: to define a ramp event, we have to determine values for its three key characteristics: direction, duration and magnitude (see Figure 1). With respect to direction there are two basic types of ramps: the upward ones (or ramp-ups), and the downward ones (or ramp-downs). The former, characterized by an increase of wind power, result from a rapid raise of wind speeds, which might (not necessarily) be due to low-pressure systems, low-level jets, thunderstorms, wind gusts, or other similar weather phenomena. Downward ramps are due to a decrease in wind power, which may occur because of a sudden depletion of the pressure gradient, or due to very high wind speeds, that lead wind turbines to reach cut-out limits (typically 22-25m/s) and shut down, in order to prevent the wind turbine from damage [4]. In order to consider a ramp event, the minimum duration is assumed to be 1 hour in [11], although in [7] these events lie in intervals of 5 to 60 minutes. The magnitude of a ramp is typically represented by the percentage of the wind farm’s nominal power - nameplate.

In [7] the authors studied the sensitivity of two ramp definitions to each one of the two parameters introduced above: ramp amplitude ranging from 150 to 600MW and ramp duration values varying between 5 and 60 minutes. The definition that we present and use in this work is similar to the one described in [7]. It is more appropriate to use in real operations since it does not considers a time-ahead point to identify a ramp event.

Definition 1 A ramp event is considered to occur at time point t , the end of an interval, if the magnitude of the increase or decrease in the power signal is greater than the threshold value, the P_{ref} :

$$|P(t) - P(t - \Delta t)| > P_{ref}$$

The parameter Δt is related to the ramp duration and defines the size of the time interval considered to identify a ramp. In [11] some

results are presented that relate this parameter to the type and magnitude of identified ramps. The P_{ref} parameter is usually defined according to the specific features of the wind farm site and, usually, is defined as a percentage of the nominal wind power capacity or a specified amount of megawatts.

A comprehensive analysis of ramp modeling and prediction may be found in [2].

Algorithm 1: SHREA: a stream-based ramp predictor

input : Three time series: \mathbb{P}_T , wind power measurements; \mathbb{O}_T , wind speed measurements; and \mathbb{J}_T , wind Speed forecasts; a , the forecast horizon; P_{ref} , threshold to identify ramp events; Δt , the ramp definition parameter; W , the PAA parameter that specifies the amount of signal aggregation; σ , a forgetting factor

output: A sequence of predictions $\mathbb{Q}_r^d \dots \mathbb{Q}_{r+a}^d$ for each period/window $d = 1, \dots$

```

countTimePeriods  $\leftarrow$  0; flag  $\leftarrow$  0; Acount  $\leftarrow$  0; Bcount  $\leftarrow$  0;
for each period/window  $d$  do
  countTimePeriods ++
  1 Preprocessing
     $\mathbb{P}_s^d \leftarrow$  fitSpline( $\mathbb{P}^d$ ),  $\mathbb{O}_s^d \leftarrow$  fitSpline( $\mathbb{O}^d$ ),  $\mathbb{J}_s^d \leftarrow$  fitSpline( $\mathbb{J}^d$ )
     $\mathbb{P}_f^d \leftarrow$  rampDef( $\mathbb{P}_s^d$ ,  $\Delta t$ ),  $\mathbb{O}_f^d \leftarrow$  rampDef( $\mathbb{O}_s^d$ ,  $\Delta t$ ),  $\mathbb{J}_f^d \leftarrow$  rampDef( $\mathbb{J}_s^d$ ,  $\Delta t$ )
     $L^d \leftarrow$  label( $\mathbb{P}_f^d$ ,  $P_{ref}$ ); // Label Data
     $\mathbb{O}_n^d \leftarrow$  znorm( $\mathbb{O}_f^d$ ),  $\mathbb{J}_n^d \leftarrow$  znorm( $\mathbb{J}_f^d$ )
     $\mathbb{O}_{str}^d \leftarrow$  SAX(PAA( $\mathbb{O}_n^d$ )),  $\mathbb{O}_{str}^d \leftarrow$  SAX(PAA( $\mathbb{J}_n^d$ ))
  2 Learn Supervised HMM
     $\pi \leftarrow$ 
    ( $\delta(L^d(r) = rampDown)$ ,  $\delta(L^d(r) = noramp)$ ,  $\delta(L^d(r) = rampUp)$ )
     $\lambda^d(A, B, \pi) \leftarrow$  LearnHMM( $\mathbb{O}_{str}^d(1, \dots, r)$ ,  $L^d(1, \dots, r)$ , Acount, Bcount)
  3 Predict Ramp Events using the learned HMM
     $\mathbb{Q}_r^d \dots \mathbb{Q}_{r+a}^d \leftarrow$  Viterbi( $\lambda$ ,  $\mathbb{O}_{str}^d(r+1, \dots, r+a)$ )
     $\lambda^d(A, B, \pi) \leftarrow$  updateHMM( $\mathbb{O}_{str}^d(r+1, \dots)$ ,  $L^d(r+1, \dots)$ )
  4 Forgetting mechanism
    if (countTimePeriods ==  $\sigma$ ) then
       $A_{count}^{aux} \leftarrow$  Acount;  $B_{count}^{aux} \leftarrow$  Bcount; flag  $\leftarrow$  1
    if (countTimePeriods mod  $\sigma$  == 0 & flag == 1) then
      Acount  $\leftarrow$  Acount -  $A_{count}^{aux}$ ; Bcount  $\leftarrow$  Bcount -  $B_{count}^{aux}$ 

```

3 Methodology developed to Forecast Ramps

In this section we present SHREA framework, a stream-based framework that uses a supervised learning strategy to obtain a HMM. SHREA continuously learns a discrete HMM on a fixed size non-overlapping moving window and, at each time period, uses the updated HMM to predict ramp events. We introduce a forgetting mechanism to forget old wind regimes and to accommodate weather global changes. The SHREA architecture has three main steps (see algorithm pseudo-code in Algorithm 1): preprocessing phase, where a ramp filter and the SAX algorithm are used to translate real valued signals into events/strings; learning phase, where a supervised strategy is used to learn a HMM; and prediction phase, where the Viterbi algorithm is used to forecast ramp events. In the following lines we describe each one of these phases.

3.1 Preprocessing In the preprocessing phase we translate the real-valued points occurring in a given time period d , i.e. occurring inside a non-overlapping fixed size window, into a discrete time-series suitable to be used at HMM learning and prediction time. First, we fit a spline to both the wind power and wind speed measurements time series obtaining, respectively, two new signals, \mathbb{P}_s^d and \mathbb{O}_s^d . We run the same procedure over \mathbb{J} time series, a wind speed forecast, and obtain \mathbb{J}_s^d . We fit splines to the original data to remove high frequencies that can be considered noisy data. Second, we run ramp definition one, presented above in Section 2, to filter the three smoothed signals and obtain three new signals: \mathbb{P}_f^d , \mathbb{O}_f^d and \mathbb{J}_f^d . These signals are wind power and speed variations, derivative alike signals, suitable to identify ramp events. Third, we use a user-defined power variation threshold, the input parameter P_{ref} value, to translate the wind power signal \mathbb{P}_f^d into a labeled time series $L^d(1, \dots, r+a)$, where 1 is the first point of the time window, r is the forecast launch time and a is the time horizon. We map each wind power variation into one of

three labels/ramp events: ramp-up, ramp-down and no-ramp. These three labels will be the three states of our HMM and the transitions will be estimated using the points of the L^d time series.

At this point we already have the data needed to estimate the transitions of the Markov process hidden in the HMM process. Now we need to transform wind speed data into a format suitable to estimate emission probabilities of the discrete HMM that we are learning. We combine Piecewise Aggregate Approximation (PAA) and SAX algorithms [9] to translate the wind speed variations into symbolic time series, more precisely. Thus, we normalize the two wind speed signals and obtain \mathbb{O}_n^d and \mathbb{J}_n^d signals. \mathbb{O}_n^d will be used to estimate the HMM emission probabilities and the \mathbb{J}_n^d will be used as the ahead observations that will be used to predict ramp events. Next, we run the PAA algorithm in each one of these signals to reduce complexity and, again, obtain smoothed signals. The degree of signal compression is the W PAA parameter that is a user-defined parameter of SHREA. This parameter is related with time point aggregation. Next, we run the SAX algorithm to map each PAA signal into string symbols. This way we obtain two discrete signals \mathbb{O}_{str}^d and \mathbb{J}_{str}^d . After the preprocessing phase we have two discrete time series, L^d and \mathbb{O}_{str}^d that will be used to learn the HMM state transitions and emissions probabilities, respectively.

3.2 Learn a Discrete HMM Here we explain how do we learn the HMM in the time period d , and then how we update it in time.

In the HMM that we learn, compactly written $\lambda(A, B, \pi)$, the state transitions, the A parameter, are associated with wind power measurements and the emissions probabilities, the B parameter, are associated with wind speed measurements. In Figure 2 we show a HMM learned by SHREA at the end of the 2010 winter. To estimate these two parameters we use the ramp labels, $L^d(1, \dots, r)$, and the wind speed measurements signals, $\mathbb{O}_{str}^d(1, \dots, r)$, and run the well-known and straightforward supervised learning algorithm described in [12]. To estimate the transition probabilities between states, the three-way matrix A , we count the transitions between symbols observed in $L^d(1, \dots, r)$ and compute the marginals to estimate the probabilities. To estimate the emission probabilities for each state, the matrix B , we count, for each state, the observed frequency of each symbol and then use state marginals to compute the probabilities. This way, we obtain the maximum likelihood estimate of both the transitions and the emission probability matrices.

We now explain how to update the model in the time. We design our framework to improve over the time with the arriving of new data. At each time period d SHREA is fed with new data and the HMM parameters are updated to include the most recent historical data. At each time period d we update the HMM parameters by counting the state transitions and state emissions coded in the current vectors $\mathbb{O}_{str}^d(1, \dots, r)$ and $L^d(1, \dots, r)$, obtaining the number of state transitions and emissions at each HMM state, the A_{count} and B_{count} . Then, we compute the marginal probabilities of each matrix and obtain the updated HMM, the model $\lambda^d(A^d, B^d, \pi^d)$ that will be used to predict ramp events. The learned HMM, λ^d , will be used to predict ramp events occurring between r and $r + a$. In the next time period (i.e. the next fixed sized time window) we will update the λ^d HMM, using this same strategy but including also the transitions and emissions of the time period d that were not used to estimate λ^d , i.e., we update A_{count} and B_{count} with the wind measurements of the time period d occurring after d 's launch time and before $d + 1$ period launch time, the r point. By using this strategy we continuously update the HMM to include both the most recent data and all old data. By using this strategy, and with the course of time, the HMM can be-

come less sensitive to new weather regimes. Thus we introduce a forgetting strategy to update the HMM using only the most recent measurements and forgetting the old data. This strategy relies on a threshold that specifies the number of time periods to include in the HMM estimation. This forgetting parameter, σ , is a user-defined value that can be set by experienced wind power technicians. Considering that at time period d we have read σ time periods and that we backup the current counts into A_{count}^{aux} and B_{count}^{aux} temporary matrices. After reading 2σ time periods we will use the following forgetting mechanism: $A_{count}^{2\sigma} = A_{count}^{2\sigma} - A_{count}^{aux}$ and $B_{count}^{2\sigma} = B_{count}^{2\sigma} - B_{count}^{aux}$. Then, we reset A_{count}^{aux} and B_{count}^{aux} equal to the updated $A_{count}^{2\sigma}$ and $B_{count}^{2\sigma}$ matrices, respectively. Next, to predict ramp events occurring in the time periods following 2σ , we will update and use the HMM parameters obtained from the $A_{count}^{2\sigma}$ and $B_{count}^{2\sigma}$ to forecast ramp events. Every time we read a number of time periods that equals a multiple of σ we apply this forgetting mechanism using the updated auxiliary matrices.

3.3 Predict Ramp Events using the learned HMM In this step we use the HMM learned in time period d , the λ^d , and the string \mathbb{J}_{str}^d , obtained from wind speed forecasts, to predict ramp events for the time points ranging from r to $r + a$. Remember that r is the prediction launch time and a is the forecast horizon.

To obtain the ramp event predictions we run the Viterbi algorithm [12]. We feed this algorithm with \mathbb{J}_{str}^d and λ^d and get the state predictions (the ramp events) $\mathbb{Q}_{r+1}^d, \dots, \mathbb{Q}_{r+a}^d$ for the time points $r + 1, \dots, r + a$ of time period d . Saying it in other way we obtain predictions for the points occurring in a non overlapping time window starting at r and with length equal to a . We will obtain the most likely sequence of states that best explains the observations, i.e., we will obtain a sequence of states $\mathbb{Q}_{r+1}^d, \dots, \mathbb{Q}_{r+a}^d$ that maximizes the probability $P(\mathbb{Q}_{r+1}^d, \dots, \mathbb{Q}_{r+a}^d | \mathbb{J}_{str}^d, \lambda^d)$.

Regarding the π parameter, we introduce a non classical approach to estimate this parameter. We defined this strategy after observing that it is almost impossible to beat a ramp event forecaster that predicts the ramp event occurring one step ahead to be the current observed ramp event. Thus, we set π to be a distribution having zero probability for all events except the event observed at launch time, the r time point. In the pseudo code we write $\pi \leftarrow (\delta(L^d(r) == rampDown), \delta(L^d(r) == noramp), \delta(L^d(r) == rampUp))$ where δ is a Dirac delta function defined by $\delta(x) = 1$, if x is *TRUE* and $\delta(x) = 0$, if x is *FALSE*.

4 Experimental Evaluation

In this section we describe the configurations, the metrics and the results that we obtain in our experimental evaluation.

Table 1: Misclassification Costs

		Observed		
		down	no	up
Predicted	down	0	10	80
	no	20	0	10
	up	100	30	0

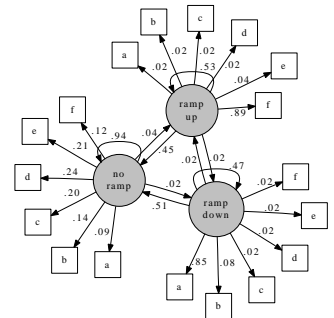


Figure 2: Winter HMM

Table 2: KSS, SS and Expected Cost Mean and standard deviation for the last 100 days of the evaluation period

Time ahead	Metric	SHREA									Persistence		
		$\Delta t=1$			$\Delta t=2$			$\Delta t=3$			$\Delta t=1$	$\Delta t=2$	$\Delta t=3$
		phE=0	phE=1	phE=2	phE=0	phE=1	phE=2	phE=0	phE=1	phE=2			
30 min	KSS	0.144(0.002)	-	-	0.332(0.001)	-	-	0.446(0.002)	-	-	0.144(0.002)	0.332(0.001)	0.446(0.002)
	SS	0(0)	-	-	0(0)	-	-	0(0)	-	-	-	-	-
	ECost	3.129(0.016)	-	-	4.176(0.027)	-	-	4.04(0.019)	-	-	3.129(0.02)	4.176(0.03)	4.041(0.02)
60 min	KSS	0.152(0.001)	0.202(0.002)	-	0.278(0.001)	0.314(0.204)	-	0.369(0.001)	0.417(0.001)	-	0.127(0.009)	0.203(0.001)	0.343(0.001)
	SS	0.028(0.001)	0.085(0.002)	-	0.094(0.00)	0.139(0.001)	-	0.038(0.001)	0.113(0.001)	-	-	-	-
	ECost	2.312(0.18)	2.107(0.014)	-	3.860(0.39)	3.719(0.39)	-	4.374(0.61)	4.108(0.61)	-	8.731(0.99)	14.687(1.50)	16.104(1.63)
90 min	KSS	0.123(0.000)	0.185(0.001)	0.231(0.002)	0.193(0.001)	0.240(0.001)	0.296(0.002)	0.271(0.001)	0.316(0.001)	0.345(0.001)	0.101(0.001)	0.163(0.002)	0.258(0.002)
	SS	0.0244(0.002)	0.093(0.001)	0.145(0.001)	0.035(0.001)	0.091(0.002)	0.159(0.001)	0.018(0.001)	0.079(0.001)	0.118(0.001)	-	-	-
	ECost	2.089(0.013)	1.938(0.012)	1.807(0.010)	4.252(0.03)	4.028(0.025)	3.728(0.024)	5.165(0.025)	4.893(0.023)	4.677(0.025)	3.204(0.030)	6.112(0.042)	6.783(0.050)

4.1 Experimental Configuration Our goal is to predict ramp events in a large-scale wind farm located in the US Midwest. To evaluate our system we collected historical data and, to make predictions, use wind speed power predictions (NWP) for the time period ranging between 3rd of June 2009 and 16th of February 2010. Each turbine in the wind farm has a Supervisory Control and Data Acquisition System (SCADA) that registers several parameters, including the wind power generated by each turbine and the measured wind speed at the turbine, the latter are 10 minute spaced point measurements. In this work we consider a subset of turbines and compute, for each time point, the subset mean wind power output and the subset mean wind speed, obtaining two time series of measurements. The wind speed power prediction for the wind farm location was obtained from a major provider. Every day we get a wind speed forecast with launch time at 6 am and having 24 hours horizon. The predictions are 10 minute spaced point forecasts. In this work we run SHREA to forecast ramp events occurring 30, 60 and 90 minutes ahead, the a parameter. We start by learning a HMM using five days of data and then use the learned, and updated, HMM to generate predictions for each fixed size non overlapping time window. Moreover, we split the day in four periods and run SHREA to learn four independent HMM models: dawn, period ranging between zero and six hours; morning, period ranging between six to twelve hours; afternoon, period ranging between twelve and eighteen hours; night, period ranging between eighteen and midnight. The last four models were only used to give some insight on the ramp dynamics and were not used to make predictions. We define a ramp event to be a change in wind power production higher than 20% of the nominal capacity, i.e., we set the P_{ref} threshold equal to 20% of the nominal capacity. Moreover, we run a set of experiments by setting Δt parameter equal to 1, 2 and 3 time points, i.e., equal to 30, 60 and 90 minutes. We run SHREA using thirty minute signal aggregation, thus each time point represents thirty minutes of data. In these experiments we also consider phase error corrections. Phase errors are errors in forecasting ramp timing [5]. We identify events that occur in a timestamp, t , not predicted at that time, but predicted instead to occur in one, or two, time periods immediately before or after t .

Furthermore, as SHREA is continuously updating the HMM, we set the forgetting parameter $\sigma = 30$, i.e., each time the system reads a new period of 30 days of data, the system forgets 30 days of old data. The amount of forgetting used in this work results from a careful study of the wind patterns.

For this configuration we compute and present the Hanssen & Kuipers Skill Score (KSS) and the Skill Score (SS) [1, 6]. Moreover, we compute the expected misclassification costs (EC) using the formula presented in [13]. The cost matrix presented in Table 1 defines the misclassification costs. We compare SHREA against a Persistence baseline algorithm. Despite its simplicity, the predictions of this model are the same as the last observation, this model is known to be hard to beat in short-time ahead predictions [10].

4.2 Results This work is twofold and here we present and analyze both the descriptive and predictive performance of the SHREA framework.

In Figure 2 we present an example of HMM generated by SHREA in February. This model was learned when running SHREA to predict 90 minutes ahead events and setting $\Delta t = 2$. This HMM has three states, each state is associated with one ramp type, and each state emits six symbols, each representing a discrete bin of the observed wind speed. The lower level of wind speed is associated with the a character and the higher level of wind speed is associated with the f character. The labels in the edges show the state emissions and the state transition probabilities.

The HMM models that we obtained in our experiments uncover interesting ramp behaviors. If we consider all the data used in these experiments, when we set $\Delta t = 1$ we found that there were detected 7% more ramp-up events than ramp-down events. When we set $\Delta t = 3$ we get the inverse behavior, we get 4% more ramp-downs than ramp-ups. This behavior is easily explained by the wind natural dynamics that causes steepest ramp-up events and smooth ramp-down events. If we analyze independently the four periods of the day we can say that we have a small number of ramp events, both ramp-ups and ramp-downs, in the afternoon. If we compute the mean number of ramps, for all Δt parameters we get approximately 30%(15%) more ramp-up(ramp-down) events at night than in the afternoon. Overall, we can say that we get more ramp events at night and, in second place, at the dawn period. Moreover, we can say that in the summer we get, both for ramp-up and ramp-down events, wind speed distributions with higher entropy, we get approximately 85% of the probability concentrated in two observed symbols. Different from this behavior, in the winter we have less entropy in the wind speed distribution associated with both types of ramp events. In the winter we have approximately 91% of the probability distribution concentrated in the one symbol. The emission probability distribution of the ramp-down state is concentrated in symbol a and the emission probability distribution in the ramp-down state is concentrated in symbol f . These two findings are consistent with our empirical visual analysis and other findings [4]: Large wind ramps tend to occur in the winter and usually there is a rapid wind speed increase followed by a more gradual wind speed decrease. These findings are also related with the average high temperature in the summer and with the stable temperatures registered during the afternoons. Considering the Δt parameter, we can say that the number of ramps, both ramp-ups and ramp-downs, increase with the Δt parameter. In general, we observe large ramps only when we compare time points that are 20 to 30 minutes apart.

As is illustrated in Figure 2 we identified a large portion of self-loops, especially ramp-up to ramp-up transitions in the winter nights. The percentage of self-loops range between 12%, when we run SHREA with $\Delta t = 1$, and 55% when we set $\Delta t = 3$. This self-loop transition shows that we have a high percentage of ramp events hav-

ing a magnitude of at least 40% of the nameplate, two times the P_{ref} threshold. Furthermore, in the winter we get a higher proportion of ramp-up to ramp-down and ramp-down to ramp-up transitions than in the summer. This is especially clear at the dawn and night periods. This phenomena can be related with the difference in the average temperatures registered in these time periods.

Before presenting the forecast performance, it must be said that the quality of ramp forecasting depends a great deal on the quality of meteorological forecasts. Moreover, as the HMMs represent probability distributions it is expected that SHREA will be biased to predict no-ramp events. Typically SHREA over predicts no-ramp events but makes less severe errors. This biased behavior of SHREA is an acceptable feature since it is better to forecast a no-ramp event when we observe a ramp-down(ramp-up) event than predicting a ramp-up(ramp-down) event. In real wind power operations (see Table 1) the cost of the later error is several times larger than the former errors.

In Table 2 we present the mean (inside brackets we present the associated standard deviation) KSS, SS and Expected Cost metrics that we obtained when running SHREA, and the reference model, to predict ramp events occurring in the last hundred days of the evaluation period.

Before presenting a detailed discussion of the obtained results, we must say that, if we consider the same Δt parameter, in all experiments we obtained better, or equal, results than the baseline algorithm, the Persistence algorithm. Moreover, we must say that when we generate predictions for the 30 minute horizon (one time point ahead, since we use 30 minutes aggregation) we get the same results as the Persistence model. This phenomena is related with the strategy that we used to define the HMM initial state distribution. Remember that we set the HMM π parameter equal to the last state observed.

As expected, the KSS results worsen with the increase of the time horizon. It is well known that the forecast reliability/fit worsens as the distance from the forecast launch time increases. Moreover we can say that we obtained better KSS values for the morning period than in the other three periods of the day. For lack of space we do not present a detailed description of the results that we obtain when we run SHREA to predict ramp events occurring in each one of the four periods of the day. This can be related with the wind speed forecasts launch time. The wind speed forecast that we use in this work is updated every day at 6 am.

The analysis of the Δt parameter shows that the mean KSS values increase with the increase in the Δt value. Again, this can be explained by the wind patterns, typically the wind speed increases smoothly during more than 30 minutes. In Table 2 we can see clearly that SHREA performance improves with the increase in Δt parameter. We observe the same behavior when inspecting the results that we obtained by running the Persistence algorithm. Concerning the SS, we can see that we obtain improvements over the Persistence forecast that ranges between 0% and 16%.

Concerning the phase error technique, we get important improvements for the two phase error parameter values considered in this study. The amount of improvement that we obtained by considering the phase error can be valuable in real time operations. The technicians can prepare the wind farm to deal with a nearby ramp event. In Table 2 we present the results without considering the phase error technique, $phE = 0$, and considering one time point (30 minutes), $phE = 1$, and two time points (60 minutes), $phE = 2$, phase errors corrections.

We also introduce a misclassification cost analysis framework that can be used to quantify the management decisions. We define a mis-

classification cost scenario (see Table 2) and show that SHREA produces valuable predictions. In this real scenario, SHREA generates significant lower operational costs and better operational performance than the baseline model.

5 Conclusions and Future Work

In this work we obtained some insights on the intricate mechanisms hidden in the ramp event dynamics and obtain valuable forecasts for very short-time horizons. For instance, we can now say that steepest and large wind ramps tend to occur more often in the winter. Moreover, typically there is a rapid wind speed increase followed by a more gradual wind speed decrease. Overall, with the obtained HMM models we both obtained insights on the wind ramp dynamics and generate accurate predictions that prove to be cost beneficial when compared against a Persistence forecast method.

The performance of SHREA is heavily dependent on the wind speed forecasts quality. Thus, in a near future we hope to get special purpose NWP suitable to detect ramp events and having more frequent daily updates. Moreover, we will study multi-variate HMM emissions to include other NWP parameters like wind direction and temperature.

Acknowledgments: This manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (“Argonne”). Argonne, a U.S. Dep. of Energy Office of Science laboratory, is operated under Contract No. DE AC02-06CH11357. The authors also acknowledge EDP Renewables, North America, LLC. This work was also funded by the ERDF - through the COMPETE programme and by National Funds through the FCT Project KDUS.

REFERENCES

- [1] K.T. Bradford, R.L. Carpenter, and B. Shaw, ‘Forecasting southern plains wind ramp events using the wrf model at 3-km’, in *AMS Student Conference*, (2010).
- [2] C. Ferreira, J. Gama, V. Miranda, and A. Botterud, ‘A survey on wind power ramp forecasting’, in *Report ANL/DIS 10-13*, Argonne National Laboratory, (2010).
- [3] U. Focken and M. Lange, ‘Wind power forecasting pilot project in alberta’, Oldenburg, Germany: energy & meteo systems GmbH, (2008).
- [4] J. Freedman, M. Markus, and R. Penc, ‘Analysis of west texas wind plant ramp-up and ramp-down events’, in *AWS Truewind, LLC*, Albany, NY, (2008).
- [5] B. Greaves, J. Collins, J. Parkes, and A. Tindal, ‘Temporal forecast uncertainty for ramp events’, *Wind Engineering*, **33(11)**, 309–319, (2009).
- [6] A.W. Hanssen and W.J.A. Kuipers, ‘On the relationship between the frequency of rain and various meteorological parameters’, *Mededelingen van de Verhandlungen*, **81**, (1965).
- [7] C. Kamath, ‘Understanding wind ramp events through analysis of historical data’, in *IEEE PES Transmission and Distribution Conference and Expo*, New Orleans, LA, United States, (2010).
- [8] A. Kusiak and H. Zheng, ‘Prediction of wind farm power ramp rates: A data-mining approach’, *J. of Solar Energy Engineering*, **131**, (2009).
- [9] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, ‘A symbolic representation of time series, with implications for streaming algorithms’, in *8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, San Diego, CA, (2003).
- [10] C. Monteiro, R. Bessa, V. Miranda, A. Botterud, J. Wang, and G. Conzelmann, ‘Wind power forecasting: State-of-the-art 2009’, in *Report ANL/DIS 10-1*, Argonne National Laboratory, (2009).
- [11] C. W. Potter, E. Gritmit, and B. Nijssen, ‘Potential benefits of a dedicated probabilistic rapid ramp event forecast tool’, *IEEE*, (2009).
- [12] L.R. Rabiner, ‘A tutorial on hidden markov models and selected applications in speech recognition’, *Proceedings of the IEEE*, **77(2)**, (1989).
- [13] A. Srinivasan, ‘Note on the location of optimal classifiers in n-dimensional roc space’, in *Oxford University Technical Report PRG-TR-2-99*, Oxford, England, (1999).