# Predicting Ramp Events with a Stream-based HMM framework

Carlos Abreu Ferreira[1], João Gama[2], Vítor Santos Costa[3], Vladimiro Miranda[4] and Audun Botterud[5]

[1] LIAAD-INESC TEC and ISEP - Polytechnic Institute of Porto, Porto, Portugal
[2] LIAAD-INESC TEC and Faculty of Economics - University of Porto, Porto, Portugal
[3] CRACS-INESC TEC and Faculty of Sciences - University of Porto, Porto, Portugal
[4] INESC TEC and Faculty of Engineering - University of Porto, Porto, Portugal
[5] Argonne National Laboratory, Argonne, IL, USA

**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 SHRED 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. SHRED 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 SHRED framework against Persistence baseline in predicting ramp events occurring in short-time horizons, ranging from 30 minutes to 90 minutes. SHRED consistently exhibits more accurate and cost-effective results than the baseline.

## 1 Introduction

Ramping is one notable characteristic in a time series associated with a drastic change in value over 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, especially if there is thermal generation dominance in the power system. Unit commitment decisions must prepare the generation schedule in order to smoothly accommodate forecasted drastic changes in wind power availability [2]. Different strategies are used to mitigate or take advantage of ramp-up and ramp-down events. In this paper we present SHRED (Stream-based Hmm Ramp Event Detector) a novel stream-based framework developed to analyze and predict ramping events in short term wind power forecasting.

The development of the SHRED framework answers 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 past wind power and speed measurements and Numerical Weather Predictions (NWP), specially wind speed predictions, to generate reliable real-time predictions? How can we continuously adapt SHRED 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, SHRED framework uses a supervised strategy whereas HMM parameters 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 (first-order differences) 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 SHRED 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 a short-time ahead window we use 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 SHRED 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]. To validate our contribution we compute the Hanssen & Kuipper's Skill Score (KSS) and the Skill Score (SS) [1, 6] of the obtained three-way contingency tables. Moreover, we define a cost-sensible metric that takes into account the cost of all misclassifications.

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 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 and a discrete HMM, learned from wind speed measurements, to predict short-time ahead ramp events.
– Introduce a slightly modified ramp definition that is suitable to be used in a stream based predictor.

In the next Section we introduce the ramp event forecast problem. In Section 3 we present a detailed description of our framework. Next, in Section 4, we present and discuss the obtained results. At the end of the paper, we give an overview of this work 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 that can be related with the location or with the wind 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 (although 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

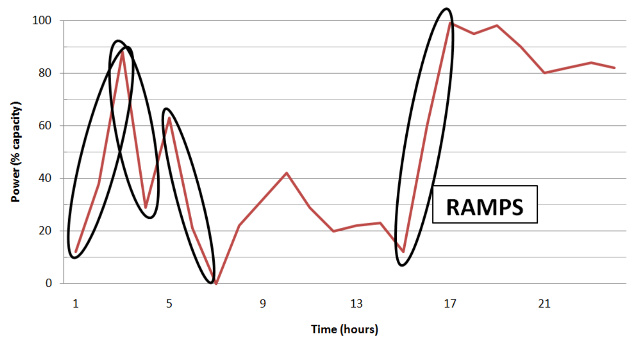


Fig. 1: Illustration of ramp events, defined as a change of at least 50% in power in an interval of 4 hours

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 this section we present one ramp event definition. As mentioned above, a ramp event can be characterized according to three features: direction, magnitude and duration. However, if we consider that ramp magnitude values range from positive to negative, then we can characterize a ramp using only magnitude and duration features. The sign of the magnitude value can give us the ramp direction: positive magnitude values correspond to upward ramps and negative magnitude values correspond to downward ramps.

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 600 MW 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 consider 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 $\Delta t$ is related to the ramp duration (given in minutes or hours) and defines the size of the time interval considered to identify a ramp. In [11, 14] some results are presented that relate this parameter to the type and magnitude of the 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 as a specified amount of megawatts.

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

**Algorithm 1**: SHRED: a stream-based ramp event predictor

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

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

$countTimePeriods \leftarrow 0$; $flag \leftarrow 0$; $A^{count} \leftarrow 0$; $B^{count} \leftarrow 0$;
**for** *each period/window* $d$ **do**
  $countTimePeriods + +$
  **Preprocessing**
  $\mathbb{P}_s^d \leftarrow$ fitSpline($\mathbb{P}^d$), $\mathbb{O}_s^d \leftarrow$ fitSpline($\mathbb{O}^d$), $\mathbb{F}_s^d \leftarrow$ fitSpline($\mathbb{F}^d$)
  $\mathbb{P}_f^d \leftarrow$ rampDef($\mathbb{P}_s^d$, $\Delta t$), $\mathbb{O}_f^d \leftarrow$ rampDef($\mathbb{O}_s^d$, $\Delta t$), $\mathbb{F}_f^d \leftarrow$ rampDef($\mathbb{F}_s^d$, $\Delta t$)
  $L^d \leftarrow$ label($\mathbb{P}_f^d$, $P_{ref}$); *Label Data*
  $\mathbb{O}_n^d \leftarrow$ znorm($\mathbb{O}_f^d$), $\mathbb{F}_n^d \leftarrow$ znorm($\mathbb{F}_f^d$)
  $\mathbb{O}_{str}^d \leftarrow$ SAX(PAA($\mathbb{O}_n^d$)), $\mathbb{OF}_{str}^d \leftarrow$ SAX(PAA($\mathbb{F}_n^d$))
  **Learning a discrete 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($O_{str}^d(1, \dots, r)$, $L^d(1, \dots, r)$, $A_{count}, B_{count}$)
  **Predicting Ramp Events using the learned HMM**
  $\mathbb{Q}_r^d \dots \mathbb{Q}_{r+a}^d \leftarrow Viterbi(\lambda, \mathbb{OF}_{str}^d(r+1, \dots, r+a))$
  $\lambda^d(A, B, \pi) \leftarrow$ updateHMM($O_{str}^d(r+1, \dots)$, $L^d(r+1, \dots)$)
  **Forgetting mechanism**
  **if** *(countTimePeriods==σ)* **then**
    $A_{count}^{aux} \leftarrow A_{count}$; $B_{count}^{aux} \leftarrow B_{count}$; $flag \leftarrow 1$
  **if** *(countTimePeriods mod σ == 0 & flag==1)* **then**
    $A_{count} \leftarrow A_{count} - A_{count}^{aux}$; $B_{count} \leftarrow B_{count} - B_{count}^{aux}$

# 3 Methodology developed to Forecast Ramps

In this section we present a detailed description of the SHRED framework, a stream-based framework that uses a supervised learning strategy to obtain a HMM. SHRED 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 global weather changes. The SHRED 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 the $\mathbb{F}$ time series, a wind speed forecast, and obtain $\mathbb{F}_s^d$. 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 $\mathbb{F}_f^d$. These signals are wind power and speed variations (first-order differences) 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 $L^d(1, \ldots, r+a)$, 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 the Piecewise Aggregate Approximation(PAA) and SAX algorithms [9] to translate the wind speed variations into symbolic time series. Thus, we normalize the two wind speed signals and obtain $\mathbb{O}_n^d$ and $\mathbb{F}_n^d$ signals. $\mathbb{O}_n^d$ will be used to estimate the HMM emission probabilities and the $\mathbb{F}_n^d$ will be used as the ahead observations that will be used to predict ramp events. Next, we run the PAA algorithm in each signal 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 SHRED. 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 $\mathbb{F}_{str}^d$. 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 Learning a Discrete HMM

Here we explain how 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 (Section 4) we show an example of a HMM learned by SHRED. To estimate these two parameters we use the ramp labels, $L^d(1, \ldots, r)$, and the wind speed measurements signals, $\mathbb{O}_{str}^d(1, \ldots, r)$, and run the well-known and straightforward supervised learning algorithm described in [12]. To estimate the transitions probabilities between states, the three-way matrix $A$, we count the transitions between symbols observed in $L^d(1, \ldots, r)$ and compute the marginals to estimate the transition 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 time. We design our framework to improve over time with the arriving of new data. At each time period $d$ SHRED 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 $O^d_{str}(1, \ldots, r)$ and $L^d(1, \ldots, r)$, obtaining the number of state transitions and emissions at each HMM state, the $A_{counts}$ and $B_{counts}$. 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 $\lambda^d$ HMM 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 $\lambda^d$ HMM, using this same strategy but including also the transitions and emissions of the time period $d$ that were not used to estimate $\lambda^d$, i.e., we update $A_{counts}$ and $B_{counts}$ with 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 become 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, $\sigma$, is a user-defined value that can be set by experienced wind power forecasters. Considering that at time period $d$ we have read $\sigma$ time periods and that we backup the current counts into $A^{aux}_{counts}$ and $B^{aux}_{counts}$ temporary matrices. After reading $2\sigma$ time periods we will use the following forgetting mechanism: $A^{2\sigma}_{counts} = A^{2\sigma}_{counts} - A^{aux}_{counts}$ and $B^{2\sigma}_{counts} = B^{2\sigma}_{counts} - B^{aux}_{counts}$. Then, we reset $A^{aux}_{count}$ and $B^{aux}_{counts}$ equal to the updated $A^{2\sigma}_{counts}$ and $B^{2\sigma}_{counts}$ matrices, respectively. Next, to predict ramp events occurring in the time periods following $2\sigma$, we will update and use the HMM parameters obtained from the $A^{2\sigma}_{counts}$ and $B^{2\sigma}_{counts}$ to forecast ramp events. Every time we read a number of time periods that equals a multiple of $\sigma$ we apply this forgetting mechanism using the updated auxiliary matrices.

### 3.3 Predicting Ramp Events using the learned HMM

In this step we use the HMM learned in time period $d$, the $\lambda^d$, and the string $\mathbb{F}^d_{str}$, 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 $\mathbb{F}^d_{str}$ and $\lambda^d$ and get the state predictions (the ramp events) $\mathbb{Q}^d_{r+1}, \ldots, \mathbb{Q}^d_{r+a}$ for the time points $r + 1, \ldots, r + a$ of time period $d$. In other words, 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 $\mathbb{Q}^d_{r+1}, \ldots, \mathbb{Q}^d_{r+a}$ that maximizes the probability $P(\mathbb{Q}^d_{r+1}, \ldots, \mathbb{Q}^d_{r+a} | \mathbb{F}^d_{r+1}, \ldots, \mathbb{F}^d_{r+a}, \lambda^d)$.

Regarding the $\pi$ 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 $\pi$ 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 $\pi \leftarrow (\delta(L^d(r) == ramp - down), \delta(L^d(r) == no - ramp), \delta(L^d(r) == ramp - up))$ where $\delta$ 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.

### 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 SHRED 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 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 SHRED 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 hour 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 $\Delta t$ parameter equal to 1, 2 and 3 time points, i.e., equal to 30, 60 and 90

Table 1: Misclassification Costs

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

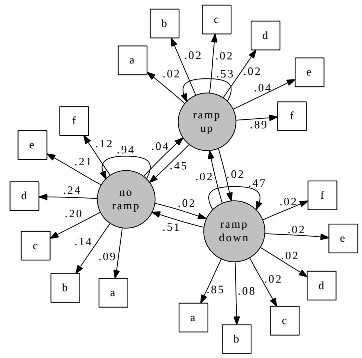


Fig. 2: HMM learned in the Winter of 2010

minutes. We run SHRED 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 SHRED is continuously updating the HMM, we set the amount of forgetting, the $\sigma$ parameter equal to 30 days of data, 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 & Kuipper's 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 SHRED 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 Metrics for Ramp Event Detection

The Hanssen & Kuipper's Skill Score (KSS), also known as Pierce's Skill Score or the True Skill Score, is a widely used [1, 6] metric that takes into account all the elements of the contingency table. It measures the forecast accuracy in predicting correct events relative to a random chance forecast.

$$KSS = \frac{\sum_{i=1}^{r} p(f_i, o_i) - \sum_{i=1}^{r} p(f_i)p(o_i)}{1 - \sum_{i=1}^{r} (p(f_i))^2}$$

The KSS takes values in the interval $[-1; 1]$, where 0 indicates no skill, 1 is the perfect score and the negative values are associated with a *perverse* forecast system.

To access the comparative performance of SHRED against the reference algorithm, the Persistence algorithm, we compute the Skill Score:

$$SS = \frac{KSS_{SHRED} - KSS_{Persistence}}{1 - KSS_{Persistence}}$$

Where $KSS_{SHRED}$ and $KSS_{Persistence}$ are the KSS metric values that we obtained, respectively, for our framework and Persistence.

These two metrics are used to access the hit performance of our methodology but give few or no information about the costs of using the SHRED model to manage scheduling and dispatch decisions in the power grid. Thus, we compute the expected misclassification costs (EC) using the formula presented in [13] and the cost matrix $C$ presented in Table 1:

$$EC = \sum_{i,j,i \neq j} \pi(j)C(i|j)P(i|j).$$

where $\pi(j)$ is the prior class probability of class $j$, $C(i|j)$ is the cost of misclassifying an event of class $j$ as an event of class $i$ and $P(i|j)$ is the proportion of examples from events of class $j$ that are predicted as class $i$ by the HMM. If we set $k$ to be the number of classes, for each $j$ we get $\sum_{i=1}^{k} P(i|j) = 1$.

### 4.3 Results

This work is twofold and here we present and analyze both the descriptive and predictive performance of the HMM models generated by SHRED.

In Figure 2 we present an example of the HMM generated by SHRED in February. This model was learned when running SHRED to predict 90 minutes ahead events and setting $\Delta t = 2$. This HMM has three states, each state is associated with one ramp event 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 emission and transition probabilities of each state. For each fixed size non overlapping moving time window the SHRED system generates a HMM describing the most recent ramp behavior. Despite the prediction methodology that we introduce in this work, the wind power operator can get some insights on the past and forthcoming ramp events only by inspecting the stream of HMMs generated by the SHRED framework.

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-dow 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

Table 2: KSS mean and standard deviation values for the last 100 days of the evaluation period

| | SHRED Mean Generalization Error and Std. Deviation | | | | | | | | | Persistence | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\Delta t$=1 | | | $\Delta t$=2 | | | $\Delta t$=3 | | | $\Delta t$=1 | $\Delta t$=2 | $\Delta t$=3 |
| Time ahead | phE=0 | phE=1 | phE=2 | phE=0 | phE=1 | phE=2 | phE=0 | phE=1 | phE=2 | | | |
| 30 min | 0.144 (0.002) | – | – | 0.332 (0.001) | – | – | 0.446 (0.002) | – | – | 0.144 (0.002) | 0.332 (0.001) | 0.446 (0.002) |
| 60 min | 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) |
| 90 min | 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) |

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 $\Delta t$ 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 one symbol. The emission probability distribution of the ramp-down state is concentrated in symbol $a$ and the emission probability distribution in the ramp-up 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 $\Delta t$ parameter, we can say that the number of ramps, both ramp-ups and ramp-downs, increase with the $\Delta t$ parameter. This is easily explained when we observe the wind speed measurements signal. Generally, 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 SHRED 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 having a magnitude of at least 40%, 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 dawn and night periods. This phenomenon needs a deeper investigation but it can be related with the difference in the average temperatures registered in these time periods.

Table 3: Skill Score mean and standard deviation values for the last 100 days of the evaluation period

| Time ahead | SHRED Mean Generalization Error and Std. Deviation $\Delta t$=1 phE=0 | $\Delta t$=1 phE=1 | $\Delta t$=1 phE=2 | $\Delta t$=2 phE=0 | $\Delta t$=2 phE=1 | $\Delta t$=2 phE=2 | $\Delta t$=3 phE=0 | $\Delta t$=3 phE=1 | $\Delta t$=3 phE=2 |
|---|---|---|---|---|---|---|---|---|---|
| 30 min | 0 (0) | – | – | 0 (0) | – | – | 0 (0) | – | – |
| 60 min | 0.028 (0.001) | 0.085 (0.002) | – | 0.094 (0.00) | 0.139 (0.001) | – | 0.038 (0.001) | 0.113 (0.001) | – |
| 90 min | 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) |

Table 4: Expected Cost mean and standard deviation values for the last 100 days

| Time ahead | SHRED Mean Generalization Error and Std. Deviation $\Delta t$=1 phE=0 | $\Delta t$=1 phE=1 | $\Delta t$=1 phE=2 | $\Delta t$=2 phE=0 | $\Delta t$=2 phE=1 | $\Delta t$=2 phE=2 | $\Delta t$=3 phE=0 | $\Delta t$=3 phE=1 | $\Delta t$=3 phE=2 | Persistence $\Delta t$=1 | Persistence $\Delta t$=2 | Persistence $\Delta t$=3 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 30 min | 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 | 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 | 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) |

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 SHRED will be biased to predict no-ramp events. In Table 5 we present a three-way contingency table that illustrates SHRED prediction pattern. In Table 6 we present the three-way contingency table that summarizes the results obtained by the Persistence model. Typically SHRED over predicts no-ramp events but makes less severe errors. This biased behavior of SHRED 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 latter error is typically several times larger than the former errors.

Table 5: SHRED three-way contingency table obtained to predict 90 minutes ahead ramp events, and using $\Delta t = 2$. In this experiment we use all the data.

| Predicted | Observed down | Observed no | Observed up |
|---|---|---|---|
| down | 169 | 414 | 40 |
| no | 735 | 8644 | 735 |
| up | 55 | 407 | 189 |

Table 6: Persistence three-way contingency table obtained to predict 90 minutes ahead ramp events, and using $\Delta t = 2$. In this experiment we use all the data.

| Predicted | Observed down | Observed no | Observed up |
|---|---|---|---|
| down | 185 | 693 | 118 |
| no | 611 | 8112 | 631 |
| up | 163 | 660 | 215 |

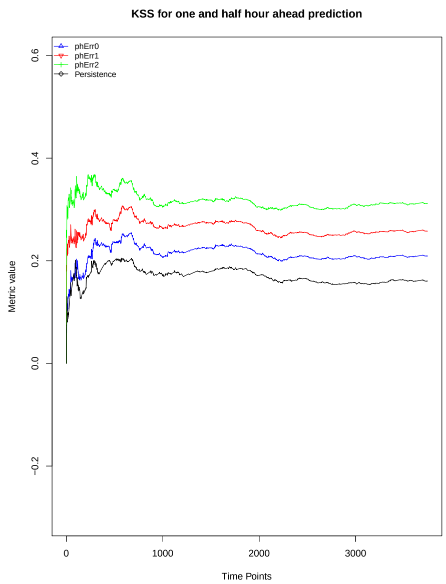


Fig. 3: KSS for 90 min ahead

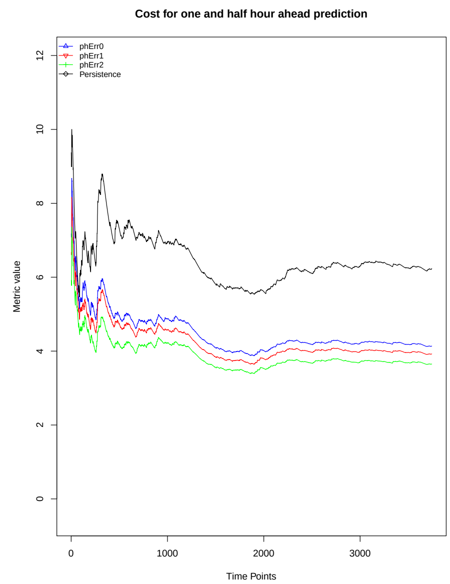


Fig. 4: Expected Cost for 90 min ahead

In Tables 2, 3 and 4 we present the mean (inside brackets we present the associated standard deviation) KSS, SS and Expected Cost metrics, respectively, that we obtained when running SHRED, and the reference model, to predict ramp events occurring in the last hundred days of the evaluation period. In Figures 3 and 4 we present the typical KSS and Expected Cost curves when we use the entire dataset.

Before presenting a detailed discussion of the obtained results, we must say that in all experiments, if we consider the same $\Delta t$, 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 $\pi$ parameter equal to the last state observed.

As expected, the KSS results worsen with the increase of the time horizon. This is a typical behavior that can be explained by wind speed forecast reliability. 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. Due to space limitations, we do not present a detailed description of the results that we obtain when we run SHRED 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 $\Delta t$ parameter shows that the mean KSS values increase with the increase in the $\Delta t$ value. Again, this can be explained by the wind patterns, typically the wind speed increases smoothly during more than 30 minutes. Moreover, the number of misclassification errors decrease with the increase in the $\Delta t$ value. In Table 2 we can see clearly that SHRED performance improves with the increase in $\Delta t$ 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 values considered in this study. The amount of improvement that we obtained by considering the phase error can be valuable in real time operations. Operators can prepare the wind farm to deal with a nearby ramp event. In Tables 2, 3 and 4 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 error corrections.

We also introduce a misclassification cost analysis framework that can be used to quantify the management decisions. We define a misclassification cost scenario (see Table 1) and show that SHRED produces valuable predictions. In this real scenario, SHRED generates significant lower costs and better operational performance than the baseline model (see Table 4). In Figure 4 we show the Expected Cost curve obtained by running SHRED and Persistence to predict 90 minutes ahead ramp events and by setting $\Delta t = 2$.

## 5 Conclusions and Future Work

In this paper we have presented SHRED, a framework developed to study and predict wind ramp events occurring in large-scale wind farms.

Based on a case study of a wind farm located in the US Midwest, we obtained some insights on the intricate mechanisms hidden in the ramp event dynamics and obtain valuable forecasts for short-time horizons. For instance, we found that the steepest and largest 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 proposed HMM models we both obtained insights into the wind ramp dynamics and we generated accurate predictions that proved to be cost beneficial when compared against a Persistence forecast method.

The performance of SHRED is heavily dependent on the wind speed forecasts quality. Thus, in the near future we hope to get special purpose NWP with frequent updates (several times a day) more suitable to detect ramp events. This way it will be possible to generate more trustworthy predictions and explore daily weather regimes in more detail. 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. Department of Energy Office of Science laboratory, is operated under Contract No. DE AC02-06CH11357. The authors also acknowledge EDP Renewables, North America, LLC, for providing the wind farm data used in the analysis. This work is also funded by the ERDF - through the COMPETE programme and by National Funds through the FCT Project KDUS. Carlos Abreu Ferreira is financially supported by the Portuguese Polytechnic Institute of Porto.

## References

1. Bradford, K., Carpenter, R., Shaw, B.: Forecasting southern plains wind ramp events using the wrf model at 3-km. In: AMS Student Conference (2010)
2. Ferreira, C., Gama, J., Matias, L., Botterud, A., Wang, J.: A survey on wind power ramp forecasting. Tech. Rep. ANL/DIS 10-13, Argonne National Laboratory (2010)
3. Focken, U., Lange, M.: Wind power forecasting pilot project in alberta. Oldenburg, Germany: energy & meteo systems GmbH (2008)
4. Freedman, J., Markus, M., Penc, R.: Analysis of west texas wind plant ramp-up and ramp-down events. In: AWS Truewind, LLC. Albany, NY (2008)
5. Greaves, B., Collins, J., Parkes, J., Tindal, A.: Temporal forecast uncertainty for ramp events. Wind Engineering 33(11), 309–319 (2009)
6. Hanssen, A., Kuipers, W.: On the relationship between the frequency of rain and various meteorological parameters. Mededelingen van de Verhandlungen 81 (1965)
7. Kamath, C.: 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. Kusiak, A., Zheng, H.: Prediction of wind farm power ramp rates: A data-mining approach. Journal of Solar Energy Engineering 131 (2009)
9. Lin, J., Keogh, E., Lonardi, S., Chiu, B.: 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. Monteiro, C., Bessa, R., Miranda, V., Botterud, A., Wang, J., Conzelmann, G.: Wind power forecasting: State-of-the-art 2009. Tech. Rep. ANL/DIS 10-1, Argonne National Laboratory (2009)
11. Potter, C.W., Grimit, E., Nijssen, B.: Potential benefits of a dedicated probabilistic rapid ramp event forecast tool. IEEE (2009)
12. Rabiner, L.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE 77(2) (1989)
13. Srinivasan, A.: Note on the location of optimal classifiers in n-dimensional roc space. Tech. Rep. PRG-TR-2-99, Oxford University (1999)
14. Zack, J., Young, S., Cote, M., Nocera, J.: Development and testing of an innovative short-term large wind ramp forecasting system. In: Wind Power Conference & Exhibition. Dallas, Texas (2010)