5-1-2018

Detection and Modeling of Wind Ramp Events in Smart Grid

Xingbang Du
Lehigh University, xingbangdu@outlook.com

Detection and Modeling of Wind Ramp Events in Smart Grid

by

Xingbang Du

Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Industrial and Systems Engineering

Lehigh University

April 2018

This thesis is accepted and approved in partial fulfillment of the requirements for the Master of Science.

Date

Boris Defourny, Thesis Advisor

Theodore K. Ralphs, Interim Chairperson of Department

Contents

List of Tables

List of Figures

Abstract

Wind ramp events have a significant influence of uncertainty in wind power production. In order to build an efficient decision-making systems for the smart grid, developing statistical models based on analysis of historical data of wind ramp events is indispensable. In this paper, we design a detection algorithm to analyze historical data, build distribution models to predict and simulate wind ramp events. Phase-type distribution consists of a convolution of the Exponential distribution which can be used to apply Markov decisions process and identify the factors which can cause wind ramp events. We use three types of Phase-type distribution to fit the data sets of duration, obtain the optimal number of phases and the parameters. Both the model of simulation and Phase-type distribution can be used to help making decisions and improving the accuracy of forecast for wind power production in smart grid.

Chapter 1

Introduction

In the wind power production, uncertainty is a problem which has a significant influence on the electricity market design, because it may result in significant cost in the grid [4] [5] [7]. An efficient decision-making system can reduce the cost of the influence from uncertainty. To build such a system, accurate statistical models for wind power production are needed [2].

Uncertainty of wind power has multiple forms and time scale with different statistical models and influence on the grid. For the short term trend, the time scale range is from seconds to minutes; for the long term trend, the time scale range is from minutes to days [8].

Wind ramp events are common with a large positive or negative power change in short time. There are two types of wind ramp events: up ramp events and down ramp events [14]. Up-ramp events consist of large positive power changes in a short period. They occur because of low levels of jets, strong low-pressure systems, thunderstorms, gusts or other weather phenomena. Down-ramp events consist of large negative power changes in a short time. They are caused by the reduction or reversal of these physical processes [7].

To build the statistical models for wind ramp events, we need to detect all the events with their parameters in given data sets. A detected ramp satisfies three conditions: minimum ramp rate, minimum start magnitude of power change and minimum end rate. According to the three conditions, we adopt those three rules as the logical basis of detection algorithm. With the algorithm, we detect all the ramp events from 2011 to 2017 with their parameters: time interval between two events, ramp duration and ramp slope.

With the data sets of time interval, ramp duration and ramp slope, we use different continuous distribution models, with quantile-quantile plot method and optimization methods, and we find the optimal statistical model for ramp events. In this model, the distribution model for both time interval and durations is a Gamma distribution, for ramp slope is an Exponential distribution.

With this model, given 3000 time points and start power point as same as actual wind power production in Spring 2017, we do the simulation of prediction of wind ramp events , then compare the result with the actual power production. We attend the "2018 ISE Department Undergraduate & Masters Research Symposium" with a poster that introduces the data analysis, modeling and simulation part. The poster is shown in Appendix A.

Based on the result of detection and modeling, we use three types of Phase-type distribution models which are Erlang distribution, Hyper-exponential distribution and Coxian Phase-type distribution to fit the data sets of duration and find the optimal number of phases.

Contributions and organization

The contributions of this work are as follows. First we design a detection algorithm with dynamic programming structure and get the data sets of parameters of detected wind ramp events. Second, we construct a compound model to predict and simulate wind ramp events. Third, we use three types of Phase-type distribution models to fit data sets of duration, find the best number of phases. The parameters and number of phases can be used to help making decisions in the smart grid and improve the accuracy of forecast for wind power production. We write a MATLAB package which includes code of wind ramp detection, modeling and simulation. It can be used to build a detection system with user interface. It allows users choose parameters and datasets themselves, get detection result, a model of wind ramp events and simulation of prediction.

The thesis is organized as follows. Chapter 2 reviews previous work on the topic. Chapter 3 describes the definition of wind ramp event and the detection algorithm, lists the detection results. Chapter 4 build some statistical and distribution models to characterize ramp events. Chapter 5 fits the data sets of duration to three types of Phase-type distributions and finds the optimal number of phases. Chapter 6 presents conclusions.

Chapter 2

Related Work

In the field of wind power production forecast, ramp events detection and modeling, [4] discuss the impact of wind power integration costs and grid integration studies on the grid, then evaluate some grid planning with operational changes that may need to be incorporated into higher levels of wind power. [2] explore the sensitivity of optimal expected profit to uncertainty in the underlying wind process. [5] present a methodology which quantifies the reserve needed on a system taking into account the uncertain nature of the wind power.

As a kind of typical event of uncertainty in wind power production, wind ramp events has several unique characteristics and causes. [7] present an overview of current ramp definitions and state-of-the-art approaches in ramp event forecasting .

Based on characteristics and causes of wind ramp events, [14] use an optimal detection technique to identify wind ramps for large time serie and make an extensive statistical analysis on the detection result.

In the field of application of Phase-type distribution, there are several special cases which are used in different research fields. The Coxian Phase Type distribution are used in the application of heath care management. [3] introduce the definition of Coxian phase-type distribution. [11] use Coxian phase-type distribution for modelling patient duration of stay in hospital and identify common characteristics of different groups of patients divided by length of stay in hospital.

Hyper-exponential distribution are used in analysis of Internet traffic's distribution models in communication network. [6] analyze network performance models by fitting mixtures of exponentials to long-tail distributions.

Erlang distribution, is a special case of both Phase-type distribution and Gamma distribution. [1] use EM Algorithm to fit Erlang, Weibull and Log normal distributions with Phase-type distributions.

Chapter 3

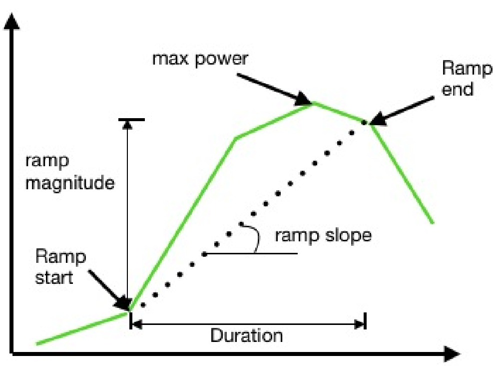
Ramp detection

In this chapter, we design an algorithm with dynamic programming structure according to three rules based on the definition of wind ramp event. With this algorithm, we detect all the wind ramp events in 2012 to 2016 then do some statistics on these parameters.

3.1 Definition of wind ramp event

Wind ramp event means that there is a large positive or negative wind power change in a short time period. It can be described by three parametersramp slope, duration and ramp magnitude. Duration is the time period of the ramp event; ramp magnitude is the power change in a ramp event; ramp slope is the rate of ramp event which equals to ratio of magnitude to duration.
The definition is also shown in Figure 3.1.

Figure 3.1: Definition of ramp events



3.2 Detection rules and algorithm

The data of wind power production is from The Bonneville Power Administration which is a non-profit federal power marketing administration based in the Pacific Northwest. We choose data from 2011 to 2016 with 632448 time points in the data sets, the minimum time period equals to 5 minutes.

To identify and detect a ramp event, we need to know when the ramp event starts and when the ramp event ends. So there are three variables for detection: start rate, start magnitude and end rate.

The rate equals to the absolute value of the ratio of magnitude to duration. The start magnitude equals to the absolute value of the power change in first 5 minutes. The end rate equals to the absolute value of ratio of current points power to largest power in the ramp events.

According to the three variables, we have three rules in detection of wind ramp events. R represents the rule sets, $R(i, j)$ represents a rule used in a time interval (i, j) [10] [7].
The three rules can be used to identify a ramp events:

$$R_1(i,j) = \begin{cases} 1 & p_j - p_i > PSW \\ 0 & p_j - p_i <= PSW \end{cases} \tag{3.1}$$

$$R_2(i,j) = \begin{cases} 1 & \frac{p_j - p_i}{t_j - t_i} > \alpha \\ 0 & \frac{p_j - p_i}{t_j - t_i} <= \alpha \end{cases} \tag{3.2}$$

$$R_3(i,j) = \prod_{m=i}^{j} 1_{\{p_{\max}\} > \beta \max((p_i \ldots) p_m)} \tag{3.3}$$

Eq.(3.1) checks if the start magnitude is larger than a given magnitude PSW, otherwise its not a start of a wind ramp event.

Eq.(3.2) checks if the rate is larger than a given minimum rate α, otherwise its not a start of a wind ramp event.

Eq.(3.3) checks if the end rate is smaller than a given rate β, otherwise its not the end of a wind ramp event [14].

With these 3 rules, we designed an algorithm in MATLAB to detect wind ramp events with data from BPA control area:

```
Ramp Detection Algorithm
N←length(p)   N is the number of data points of the wind power
For i = 1 → N
    For i + 1 = 1 → N do
        if p_j − p_i > psw then
            a wind ramp event start
                if (p_j − p_i)/(t_j − t_i) > α then
                    if p_j < β p_max then
                        the wind ramp event end at time j − 1
                    else
                        the wind ramp event did not end
                    end if
                else
                    the wind ramp has ended at time j − 1
                end if
        end if
    end for
end for
```

There are two loops in this algorithm and each of them has N iterations, so the complexity of this algorithm is $O(N^2)$. The code of detection algorithm is presented in Appendix B.

3.3 Result of Detection

A. Data Description

The data of wind power production is from The Bonneville Power Administration control area includes Oregon state and Washington state. BPA records every 5 minutes' wind power production from 2007 to present. We choose data sets from 2011 to 2016 with 632448 data points as training data, data sets of 2017 with 105408 data points as testing data.

B. Detect Result

With the algorithm, we detect wind ramp events year by year in MATLAB. We got the event visualization and four parameters which can describe wind ramp events: time interval between two events, ramp duration and ramp slope.

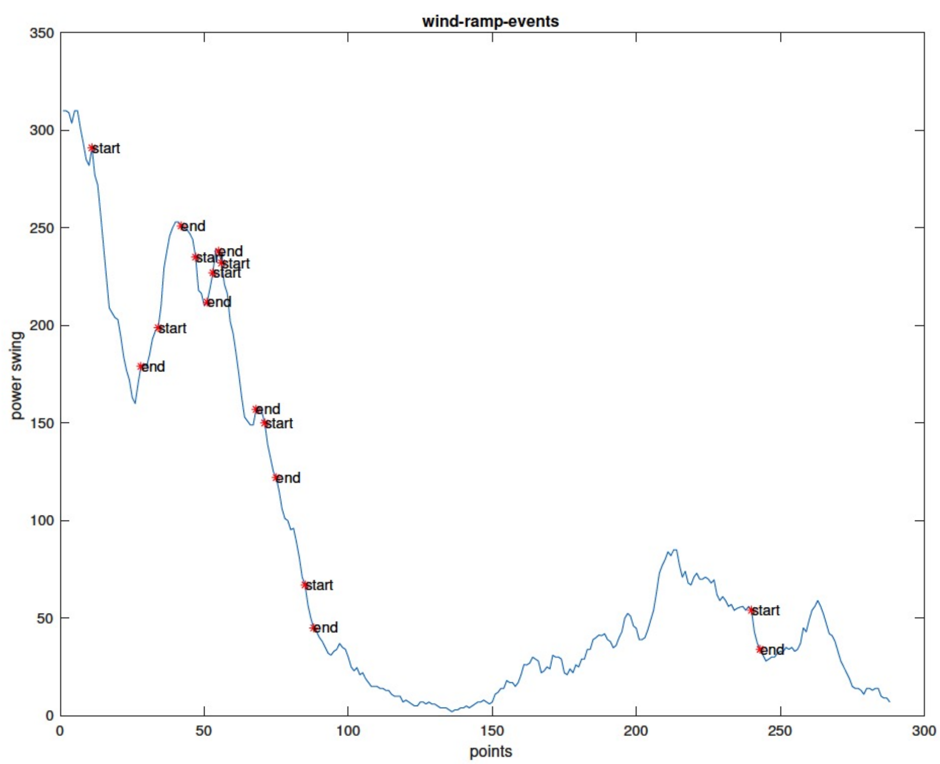
Time interval: Time period between the end of last ramp event and the start of next ramp event.

Ramp duration: Time period between start point and end point of a ramp event.

Ramp slope: Ratio of magnitude of power change to ramp duration.

For example, given $\alpha = 1.5, \beta = 0.75, psw = 10$, with data of 01/01/2011, we can see the detection result in Figure 3.2:

Figure 3.2: Detection result for 01/01/2011

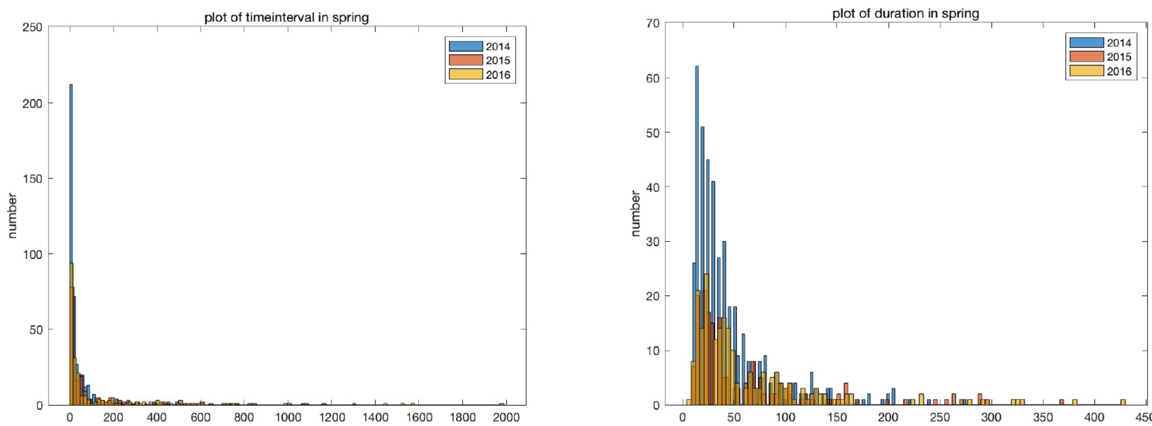


And we have the statistics of the three parameters.

For example, given $\alpha = 8.5, \beta = 0.9, psw = 100$, with data of March to Mar in 2014, 2015 and 2016,

(1) we can see the statistics of time intervals and duration in Figure 3.3:

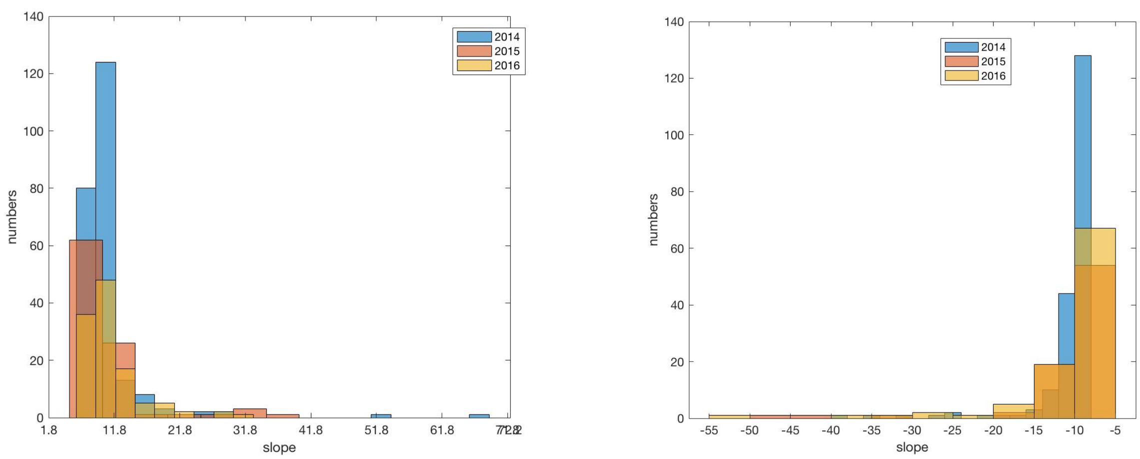
Figure 3.3: Statistics of time intervals and duration



(a) Time interval (b) Duration

(2) we can see the statistics of slope of up ramp and down ramp in Figure 3.4:

Figure 3.4: Statistics of slope of up ramp and down ramp



(a) Up Ramp (b) Down Ramp

Based on the data sets from BPA,with the algorithm, we have got the data of all ramp events with their time intervals, duration and slope. So in next chapter, based on these samples, we will try to build model and simulate ramp events.

Chapter 4

Modeling and Simulation

In this chapter, we use several distribution models, with maximum likelihood estimation and quantile-quantile plot methods, to fit data of time intervals, duration and slope, then build a combination model of wind ramp events then simulate wind ramp events in a given time.

4.1 Methodology

Maximum likelihood estimation is a parameter estimation tool for many statistical modeling techniques, especially nonlinear modeling for non-normal data [13]. The maximum likelihood estimation provides a method for evaluating model parameters using given observation data, defined models and unknown parameters. Through several experiments, observation results, using the test results to obtain a certain parameter value can make the probability of the largest number of samples, that is, the maximum likelihood estimation. In the maximum likelihood estimation, we first establish a likelihood function, then obtain the likelihood equation. By solving this equation, we get the maximum likelihood estimate, which is the parameter of the distribution model [13]. In MATLAB, we use the function $fitdist(x, distname)$ which is fitting probability distribution object to data with the principle of maximum likelihood estimation.

A quantile to quantile plot is a method which is used to compare two distribution models by plotting their quantiles against each other [16]. The QQ plot are commonly used to compare a data set to a theoretical model [15].The QQ plot sorts the sample data values from the smallest to the largest, then compares these values with the expected value from the given distribution for each quantile in the sample data. The quantile value of samples appears along the y-axis, the expected value of the specified distribution appears along the x-axis. If the result is linear, the sample data may come from the given distribution [12]. In MATLAB, we use the function $qqplot(x, pd)$ to find out if the distribution model of sample x is pd.

4.2 Data Preparation

The wind power production is influenced by great quantity of reasons. The data sets of four seasons, two ramp direction and several power levels are very different. So we separate data according to three labels:

1. Seasons

We divide the data into four parts according to 4 seasons: Spring, Summer, Autumn and Winter.

2. Ramp direction

There are two types of wind ramp events: up ramp means positive power change, down ramp means negative power change. we divide data into 2 parts according to ramp directions.

2.Power levels

When the current wind power level is high, the parameters of wind ramp events are much different from parameters when the power level is low. So, we divide the data sets into 4 parts, each part has same number of samples.

4.3 Modeling

We used Gamma distribution, Exponential distribution, Log-logistic distribution, Rayleigh distribution, Inverse Gaussian distribution and Weibull distribution, tried to fit samples of time intervals, durations and slope.

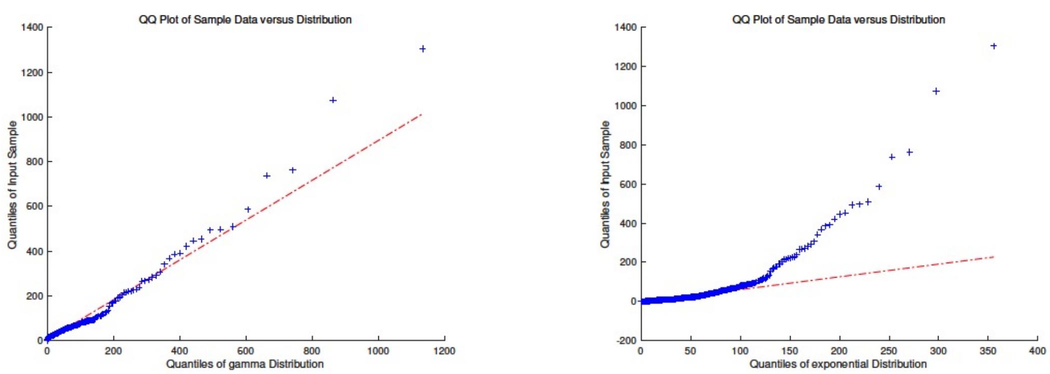
After testing with samples of different seasons, ramp directions and power levels, we find that Gamma distribution, Exponential distribution and Weibull Distribution are ideal distribution models can be used as a part of model of wind ramp events. The code of modeling is presented in Appendix B.

For example, with samples of Spring 2014, with up and down ramp direction, at power level 1 to 4, the result of fitting some ideal distribution models is as follows:

1. *Time intervals*

For the sample of time intervals, Gamma distribution and Exponential distribution are ideal distribution models, we can see the fitting result in Figure4.1:

Figure 4.1: Parameters of distribution model fitting time intervals



(a) Gamma (b) Exponential

As we can see, Gamma distribution model fits samples of time intervals better. So we choose Gamma distribution as the model of time intervals, the parameters are shown in Table 4.1:

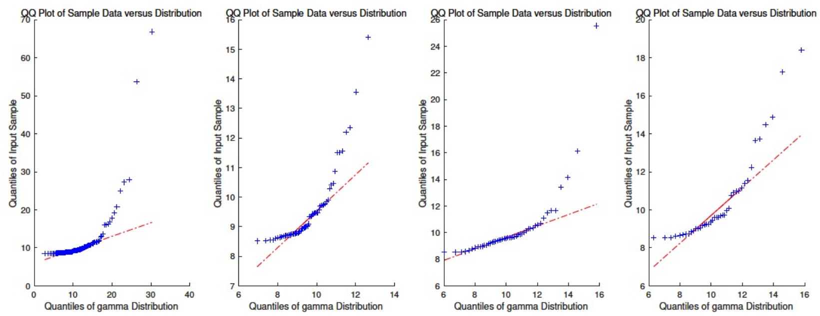
Gamma distribution
a = 0.1777 ,b = 296.8164

Table 4.1: Parameters of distribution model fitting time intervals

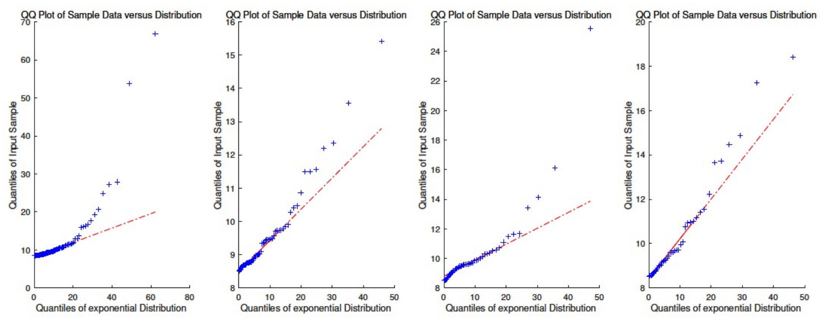
2. Slope

For the sample of slope, Gamma distribution and Exponential distribution are ideal distribution models, we can see the fitting result in Figure4.2 and Figure4.3:

Figure 4.2: Parameters of distribution model fitting slope of up ramp



(a) Gamma



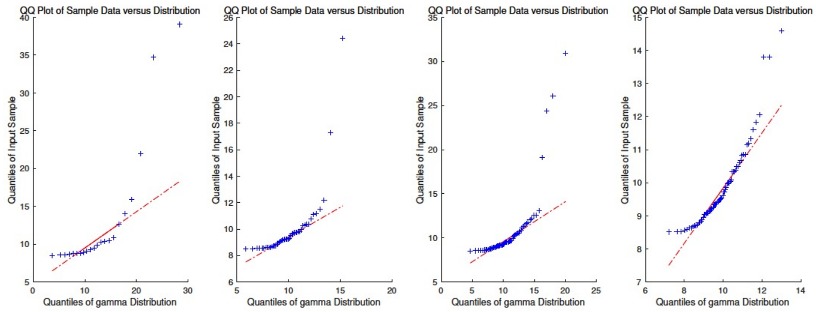
(b) Exponential

As we can see, Gamma distribution model doesn't fit samples of slope of up ramp better. So we choose Exponential distribution as the model of slope of up ramp events. The parameters are shown in Table 4.2:

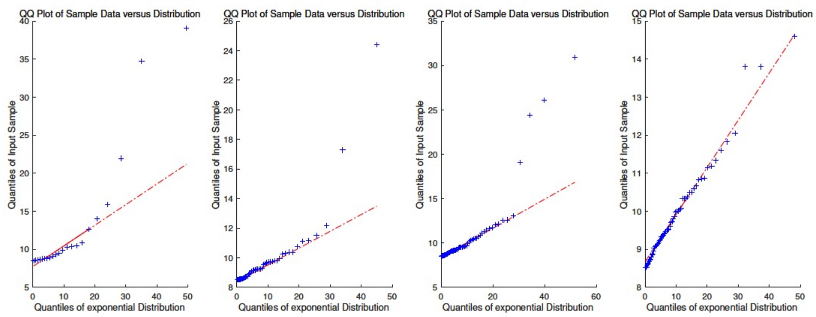
Power Level with direction	Exponential
1	12.1773
2	9.5616
3	10.2959
4	10.4811

Table 4.2: Parameters of distribution models fitting slope of up ramp

Figure 4.3: Parameters of distribution model fitting slope of down ramp



(a) Gamma



(b) Exponential

As we can see, Gamma distribution model doesn't fit samples of slope of down ramp better. So we choose Exponential distribution as the model of slope of down ramp events. The parameters are shown in Table 4.3:

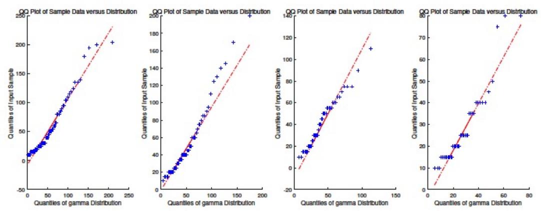
Power Level with direction	Exponential
1	13.0682
2	9.9442
3	10.8760
4	9.8630

Table 4.3: Parameters of distribution models fitting slope of down ramp

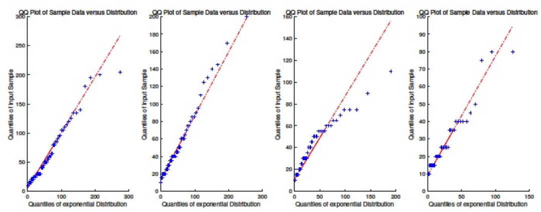
3. Duration

For the sample of slope, Gamma distributionExponential distribution and Weibull Distribution are ideal distribution models, we can see the fitting result in Figure4.4 and Figure4.5:

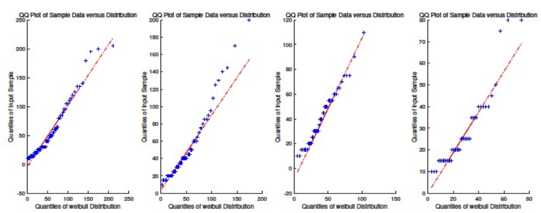
Figure 4.4: Parameters of distribution model fitting duration of up ramp



(a) Gamma



(b) Exponential



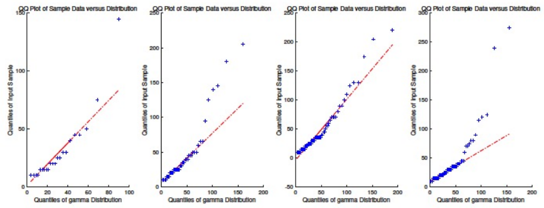
(c) Weibull

As we can see, Exponential and Weibull distribution model doesn't fit samples of duration of up ramp better. So we choose Gamma distribution as the model of duration of up ramp events. The parameters are shown in Table 4.4:

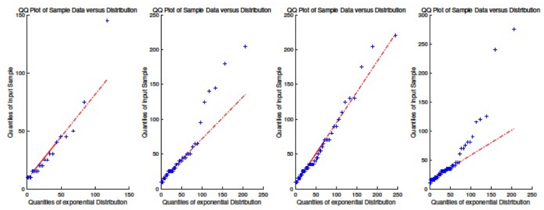
Power Level with direction	Gamma distribution
1	a = 1.6935, b=31.5887
2	a=2.1905 , b=24.1949
3	a=3.2050 , b= 12.9681
4	a=3.4290 , b=8.2865

Table 4.4: Parameters of distribution models fitting slope of up ramp

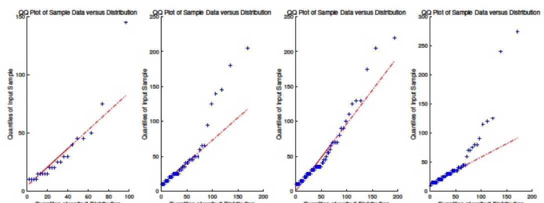
Figure 4.5: Parameters of distribution model fitting duration of down ramp



(a) Gamma



(b) Exponential



(c) Weibull

As we can see, Exponential and Weibull distribution model doesn't fit samples of duration of down ramp better. So we choose Gamma distribution as the model of duration of down ramp events. The parameters are shown in Table 4.5:

Power Level with direction	Gamma distribution
1	a = 1.9364, b= 16.0796
2	a=1.6989, b= 26.8083
3	a=1.6858 , b= 30.5802
4	a=1.7682 , b= 23.7787

Table 4.5: Parameters of distribution models fitting slope of down ramp

4.4 Simulation

Based on the result of modeling, we build a compound model for wind ramp events:

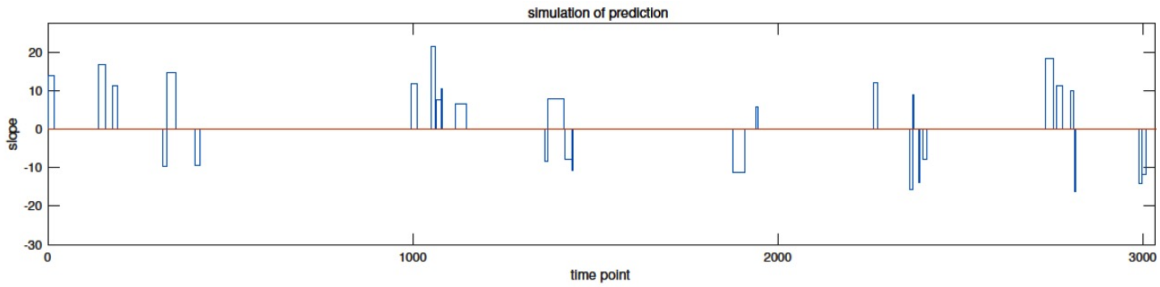
(1) The distribution for time intervals is Gamma distribution model.

(1) The distribution for durations is Gamma distribution model.

(1) The distribution for slope is Exponential distribution model.

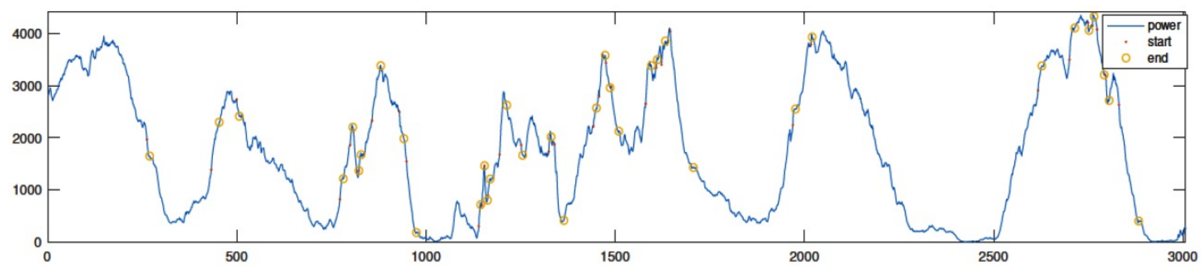
Then based on this model, we got the simulation of prediction of wind ramp events in the first 3000 time points with the same start point as Spring2017. The code of simulation is presented in Appendix B.We can see the predicted ramp events with slope and duration in Figure 4.6:

Figure 4.6: Simulation of ramp events



Then we compare the simulation with the actual wind power production with marked ramp events in the first 3000 time points in Spring2017 in Figure 4.7 :

Figure 4.7: Actual wind ramp events



As we can see, we can predict the time and parameters of a considerable part of the event. We find that Gamma distribution and Exponential distribution are ideal distribution models can be used in the compound model, calculated their Conditional Probability for next step of research:

Distribution	CDF	Conditinal Probability
Exponential	$1 - e^{-\lambda x}$	$e^{-\Delta \mathrm{t} \lambda}$
Gamma	$\gamma(\alpha, \beta \mathrm{x})/\Gamma(\alpha)$	$\frac{\Gamma(\alpha)-\int_0^{\beta.(\Delta \mathrm{t}+t)} m^{\alpha-1}.e^{-m}\, dm}{\Gamma(\alpha)-\int_0^{\beta.t} m^{\alpha-1}.e^{-m}\, dm}$

Table 4.6: Conditional probability

Chapter 5

Fitting Phase-type distribution

In this chapter, we used three types of Phase-type distribution model to fit the data sets of duration: Erlang distribution, Hyper-exponential distribution and Coxian phase-type distribution. The Phase-type distribution is a kind of probability distribution constructed by a convolution or mixture of exponential distributions.

[9]. This distribution is represented by a random variable, which indicates the time before absorption of a continues-time Markov process with finite state space {0,1,...,p} where 0 is absorbing and other states are transient [1].

5.1 Methodology

Now we use Phase-type distribution as the distribution model of duration of wind ramp events. Starting at k = 1 that the Phase-type model is an exponential distribution, with adding Phases, we calculate and compare the value of log-likelihood of each k, if there is no significant improvement with increasing number of phases, then we have find the best k. The method can be achieved by using the following likelihood function with n represents number of samples, t represents each sample of duration:

$$\sum_{i=1}^{n} \log\left(p \ \exp\left\{\mathrm{Qt}_i\right\} q\right) \tag{5.1}$$

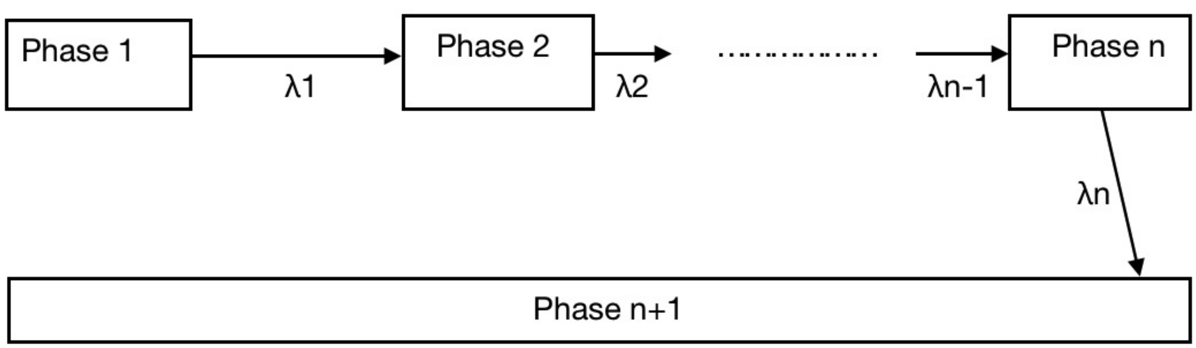
We modified the duration's data sets' time unit from minute to hour, then with interior point method which are a class of algorithms to slove linear and nonlinear convex optimization problems [12], we finished the likelihood ratio testgot the transition rate λ, μ,initial probability $p(m)$, and the value of likelihood.

5.2 Erlang Distribution fitting PH-type distribution

The Phase-type distribution has several special cases, and Erlang distribution is one of them with two or more identical phases in sequence. It is also a Gamma distribution with shape parameter k which is integer. As we described in Chapter 4, Gamma distribution is an ideal distribution model for duration, so we can use Erlang distribution's Phase-type form to fit the data sets of duration.

The P(t) described by the Phase type distribution represents the probability that the process is active at time t [11]. Let $X(t); t \geq 0$ be a continuous-time Markov chain with $n+1$ states, λ be the rate of movement from Ph_1 to Ph_2, Ph_2 to Ph_3,..., Ph_{n-1} to Ph_n, λ be the rate of movement from Ph_n to absorbing phase Ph_{n+1} (Figure5.1).

Figure 5.1: An illustration of Erlang Distribution fitting PH-type distribution



The probability density function of T is:

$$f(t) = p\ exp\{Qt\}\ q, \tag{5.2}$$

$$p = (1\ 0\ 0\ ...\ 0), \tag{5.3}$$

$$q = -Q1 = (0\ 0\ 0\ ...\ \lambda)^T, \tag{5.4}$$

and Q is the rate matrix of transition states,

$$Q = \begin{bmatrix} -\lambda & \lambda & 0 & \cdots & 0 & 0 \\ 0 & \lambda & \lambda & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\lambda & \lambda \\ 0 & 0 & 0 & \cdots & 0 & -\lambda \end{bmatrix}, \tag{5.5}$$

When the number of Phase equals to 1,

$$f(t) = \lambda^2 t e^{\lambda(-t)}, \tag{5.6}$$

When the number of Phase equals to 2,

$$f(t) = e^{-t\lambda} t \lambda \mu, \tag{5.7}$$

and so on.

We finished the likelihood ratio test then got the parameter λ and the value of likelihood as the result shown in table 5.1:

Log-likelihood	Estimation of parameters
k=1 L = -73.4736	λ= 1.1216
k=2 L = -68.2558	λ = 2.2432
k=3 L = -77.1395	λ = 3.3649
k=4 L = -91.4736	λ = 4.4865

Table 5.1: Result of Erlang Distribution fitting Phase-type Distribution

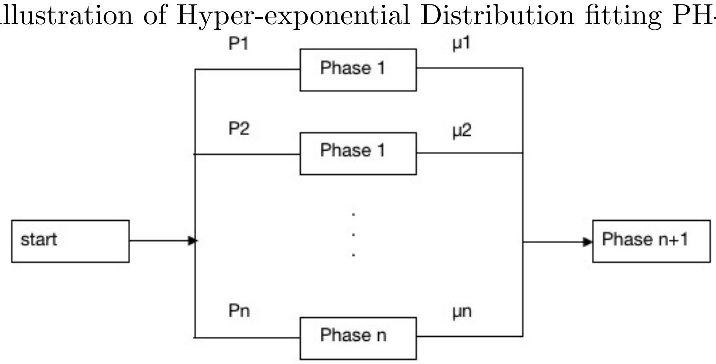
As we can see, there is no improvement of the likelihood value since K equals to 2. So, the best number of phases is 2.

5.3 Hyper-exponential Distribution fitting PH-type distribution

The Hyper-Exponential Distribution is a continuous probability distribution. It's a mixture of m exponential distributions [6]. It can be represented by as a phase type distribution with initial probability p(m).

The P(t) described by the Phase type distribution represents the probability that the process is active at time t [11]. Let $X(t); t \geq 0$ be a continuous-time Markov chain with $n+1$ states, μ_1 be the rate of movement from Ph_1 to Ph_{n+1}, μ_2 be the rate of movement from Ph_2 to Ph_{n+1},..., μ_n be the rate of movement fromPh_n to Ph_{n+1} with Ph_{n+1} is the absorb state, p_m be initial probability (Figure5.2).

Figure 5.2: An illustration of Hyper-exponential Distribution fitting PH-type distribution



The probability density function of T is:

$$f(t) = p \ exp\{Qt\} \ q, \tag{5.8}$$

$$p = (p(1) \ p(2) \ p(3) \ ... \ p(n)), \tag{5.9}$$

$$q = -Q1 = (\mu_1 \ \mu_2 \ \mu_3 \ ... \ \mu_n)^T, \tag{5.10}$$

and Q is the rate matrix of transition states,

$$Q = \begin{bmatrix} -\mu_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & -\mu_2 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\mu_{n-1} & 0 \\ 0 & 0 & 0 & \cdots & 0 & -\mu_n \end{bmatrix}, \tag{5.11}$$

When the number of Phase equals to 1,

$$f(t) = p\ \mu\ e^{-\mu\ t}, \tag{5.12}$$

When the number of Phase equals to 2,

$$f(t) = \mu_1 p_1 e^{\mu_1(-t)} + \mu_2 p_2 e^{\mu_2(-t)}, \tag{5.13}$$

and so on.

We finished the likelihood ratio test then got the parameter λ and the value of likelihood as the result shown in table 5.2:

Log-likelihood	Estimation of parameters
k=1 L = -73.4736	μ= 1.1216
k=2 L = -73.4736	$\mu_1 = 1.1216$,$\mu_2 = 1.1216$
k=3 L = -73.4819	$\mu_1 = 1.1216$,$\mu_2 = 1.1216$,$\mu_3 = 1.1216$

Table 5.2: Result of Hyper-exponential Distribution fitting Phase-type Distribution

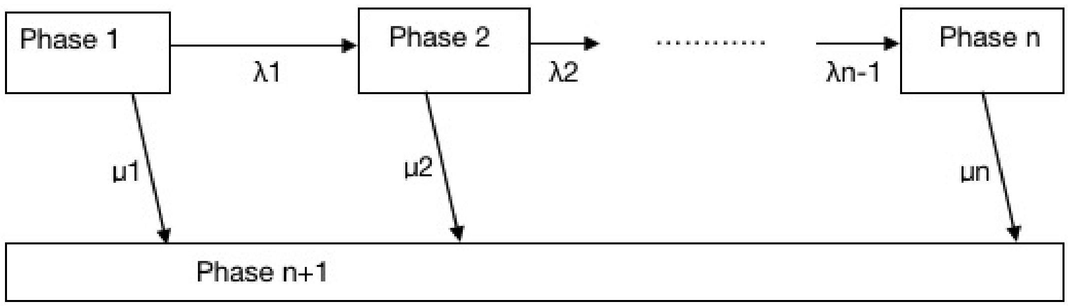
As we can see, there is no improvement of the likelihood value since K equals to 2. So, the best number of phases is 2

5.4 Coxian Phase-type Distribution

The Coxian Phase-type Distribution is a special case of Phase-type distribution which can be used to describe durations until an event happens. [3] [11].

The P(t) described by the Phase type distribution represents the probability that the process is active at time t [11]. Let $X(t); t \geq 0$ be a continuous-time Markov chain with $n+1$ states, μ_1 be the rate of movement from Ph_1 to Ph_{n+1}, μ_2 be the rate of movement from Ph_2 to Ph_{n+1},..., μ_n be the rate of movement fromPh_n to Ph_{n+1} with Ph_{n+1} is the absorb state, p_m be initial probability (Figure5.3).

Figure 5.3: An illustration of Coxian Phase-type Distribution



The probability density function of T is:

$$f(t) = p\ exp\{Qt\}\ q, \tag{5.14}$$

$$p = (1\ 0\ 0\ ...\ 0), \tag{5.15}$$

$$q = -Q1 = (\mu_1\ \mu_2\ \mu_3\ ...\ \mu_n)^T, \tag{5.16}$$

and Q is the rate matrix of transition states,

$$Q = \begin{bmatrix} -\lambda_1 - \mu_1 & \lambda_1 & 0 & \cdots & 0 & 0 \\ 0 & -\lambda_2 - \mu_2 & \lambda_2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\lambda_{n-1} - \mu_{n-1} & \lambda_{n-1} \\ 0 & 0 & 0 & \cdots & 0 & -\mu_n \end{bmatrix}, \tag{5.17}$$

When the number of Phase equals to 1,

$$f(t) = \mu_1 e^{-\mu_1} t, \tag{5.18}$$

When the number of Phase equals to 2,

$$f(t) = f(t) = \mu_1 e^{t(-\lambda_1-\mu_1)} - \frac{\lambda_1 \mu_2 e^{\mu_2(-t)} \left(e^{t(-\lambda_1-\mu_1)+\mu_2 t} - 1\right)}{\lambda_1+\mu_1-\mu_2}, \tag{5.19}$$

and so on.

We finished the likelihood ratio test then got the parameter λ and the value of likelihood as the result shown in table 5.3:

Log-likelihood	Estimation of parameters
k=1 L = -73.4736	$\mu = 1.1216$
k=2 L = -65.3219	$\lambda = 1.3630$ $\mu_1 = 0$ $\mu_2 = 3.3326$
k=3 L = -64.6676	$\lambda_1 = 3.9855$ $\lambda_2 = 1.7169$ $\mu_1 = 0$ $\mu_2 = 2.2686$ $\mu_3 = 1.1053$
k=4 L = -64.2040	$\lambda_1 = 3.5712$ $\lambda_2 = 0.9621$ $\lambda_3 = 1.6253$ $\mu_1 = 0$ $\mu_2 = 2.6091$ $\mu_3 = 0$ $\mu_4 = 1.6253$

Table 5.3: Result of fitting Coxian Phase-type Distribution

As we can see, there is no significant improvement of the likelihood value since K equals to 2. So, the best number of phases is 2.

Chapter 6

Conclusion

We have described a detection algorithm with dynamic programming recursiondemonstrate the statistics result of wind ramp events with their parameters. Based on the result, we have found accurate statistical model for wind ramp events and get the simulation of prediction. We write a MATLAB package which can be used to build a detection system with user interface. It allows users choose parameters and datasets themselves, get detection result, a model of wind ramp events and simulation of prediction.

In order to help building decision making system, we identify the optimal number of phases and parameters of three Phase type distribution models for duration which can be used to help making decisions about starting spare energy or purchasing power in the smart grid, and identify common characteristics between different groups of wind ramp events.

As future work, with the model of Phase-type distribution for duration and Exponential distribution for slope, we could employ continuous-time Markov processes. We could estimate the remaining time and the expected slope of a ramp event to find whether there exists a load shed, it will help us making decisions about starting spare energy and purchasing power in the smart grid; On other hand, we can identify common characteristics between two groups of wind ramp events, it will help to improve the accuracy of forecast for wind power production.

Bibliography

[1] Søren Asmussen, Olle Nerman, and Marita Olsson. Fitting phase-type distributions via the em algorithm. *Scandinavian Journal of Statistics*, pages 419–441, 1996.

[2] Eilyan Y Bitar, Ram Rajagopal, Pramod P Khargonekar, Kameshwar Poolla, and Pravin Varaiya. Bringing wind energy to market. *IEEE Transactions on Power Systems*, 27(3):1225–1235, 2012.

[3] David R Cox. A use of complex probabilities in the theory of stochastic processes. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 51, pages 313–319. Cambridge University Press, 1955.

[4] Jennifer DeCesaro, Kevin Porter, and Michael Milligan. Wind energy and power system operations: A review of wind integration studies to date. *The Electricity Journal*, 22(10):34–43, 2009.

[5] Ronan Doherty and Mark O'malley. A new approach to quantify reserve demand in systems with significant installed wind capacity. *IEEE Transactions on Power Systems*, 20(2):587–595, 2005.

[6] Anja Feldmann and Ward Whitt. Fitting mixtures of exponentials to long-tail distributions to analyze network performance models. *Performance evaluation*, 31(3-4):245–279, 1998.

[7] C Ferreira, Joao Gama, L Matias, Audun Botterud, and J Wang. A survey on wind power ramp forecasting. Technical report, Argonne National Laboratory (ANL), 2011.

[8] Gregor Giebel, Richard Brownsword, George Kariniotakis, Michael Denhard, and Caroline Draxl. The state-of-the-art in short-term prediction of wind power: A literature overview. *ANEMOS. plus*, 2011.

[9] Mor Harchol-Balter. *Performance modeling and design of computer systems: queueing theory in action*. Cambridge University Press, 2013.

[10] Chandrika Kamath. Associating weather conditions with ramp events in wind power generation. In *Power Systems Conference and Exposition (PSCE), 2011 IEEE/PES*, pages 1–8. IEEE, 2011.

[11] Adele H Marshall and Sally I McClean. Using coxian phase-type distributions to identify patient characteristics for duration of stay in hospital. *Health Care Management Science*, 7(4):285–289, 2004.

[12] Inc Math Works. *Matlab reference guide*. Math Works, Incorporated, 1992.

[13] In Jae Myung. Tutorial on maximum likelihood estimation. *Journal of mathematical Psychology*, 47(1):90–100, 2003.

[14] Raffi Sevlian and Ram Rajagopal. Wind power ramps: Detection and statistics. In *Power and Energy Society General Meeting, 2012 IEEE*, pages 1–8. IEEE, 2012.

[15] Henry C Thode. *Testing for normality*, volume 164. CRC press, 2002.

[16] Martin B Wilk and Ram Gnanadesikan. Probability plotting methods for the analysis for the analysis of data. *Biometrika*, 55(1):1–17, 1968.

Appendix A

Poster for Research Symposium

Simulation of prediction of Wind Ramp Events In Smart Grid

Xingbang Du and Boris Defourny
Department of Industrial & Systems Engineering, Lehigh University

Abstract

This poster is concerned with prediction and simulation of wind ramp events in smart grid. We propose an accurate wind ramp events detection algorithm, which can collect all the data of ramp events with given parameters.
We build the distribution model for wind ramp events and get simulation of prediction based on the model.

Introduction

- Uncertainty in the wind power production is a problem which has a significant influence on electricity market design and strategies.
- Wind ramp event is a typical long term trend with large positive or negative change in a short period. The frequency, scale and other properties of ramp events in wind power production are uncertain.

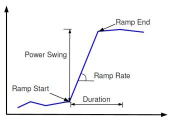


Figure 1. Definition of wind ramp event

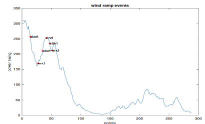


Figure 2. Example of wind ramp events

Algorithm of Detection Description

Rule of Algorithm of Detection :

- Rule 1 : R1(i, j) = 1 { pj- pi > ps } : only the difference of power of wind between 2 different points is larger than ps, point i may be the start of wind ramp.
- Rule 2 : R2(i, j) = 1 { (pj- pi)/(tj- ti) > α }: tj- ti is the duration between two points. In a wind ramp event, the rate of ramp must be larger than a given parameter α with 0 <α.
- Rule 3 : R3(i, j) = Π_m^j 1 {pm > β*max(pi,…, pm)}: there may be some drops in power of the interval, we need to check that if the wind is still in the ramp event after the drop. β is a given parameter with 0< β < 1.

Result of Detection:

- n : number of wind ramp events
- Start point: list includes start time point of each ramp event.
- Duration: list includes duration of each ramp event.
- Power swing: list includes power swing of each ramp event.

Example of result :

With α = 1.5， β = 0.75，psw = 10, data from 1/1/2011:

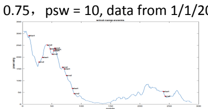


Figure 3. Example of result

Distribution Fit Data of Properties

- For α = 8.5， β = 0.9，psw = 100, data from Spring2014, Spring2015 and Spring2016,
  - Time interval in Spring 2014, 2015 and 2016 :

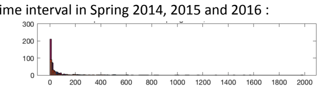


Figure 4. Time interval in Spring2014, 2015 and 2016

  - Duration in Spring 2014, 2015 and 2016:

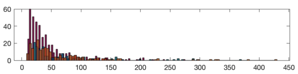


Figure 5. Duration in Spring2014, 2015 and 2016

  - Slope in Spring 2014, 2015 and 2016:

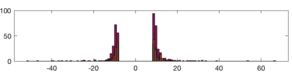


Figure 6. Slope in Spring2014, 2015 and 2016

- Distribution Fit,
  - Distribution Fit data of time intervals with Gamma distribution and Exponential distribution:

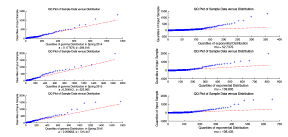


Figure 7. Distribution Fit data of time intervals in Spring2014, 2015 and 2016

  - Distribution Fit data of duration and slope with Gamma distribution and Exponential distribution:

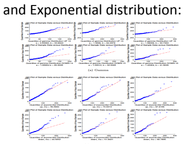


Figure 8. Distribution Fit data of duration

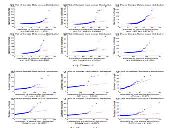


Figure 9. Distribution Fit data of slope

Distribution Model

We use Gamma distribution models to predict time intervals between wind ramp events, duration & slope of each ramp event.

- Gamma model for time interval: a = 0.2175, b = 445.7427

Probability of up evens in 7 different power levels:

Level	up
0-100	1
100-1000	0.8327
1000-2000	0.6528
2000-3000	0.5021
3000-4000	0.4424
4000-5000	0.1736
5000-∞	0

Figure 10. Probability of up evens

For up ramp events:

- Gamma model for duration: a = 1.6396, b = 40.2778
- Gamma model for slope: a = 9.6643, b = 1.1767

For down ramp events:

- Gamma model for duration: a = 1.5784, b = 33.5857
- Gamma model for slope: a = 9.0489, b = 1.3192

Simulation & Comparison

We use the distribution model to predict the slope of wind ramp events in the first 3000 time points in Spring2017:

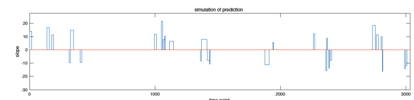


Figure 11. simulation of prediction of slope

Then we compare the simulation to the actual wind power production in the first 3000 time points in Spring2017:

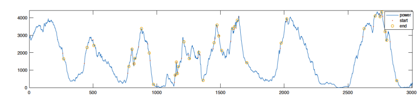


Figure 12. actual wind power production

Summary of Contributions

- Do the statistics of wind power generation data since five years.
- provide an algorithm to detect wind ramp events and their properties from the big data.
- provide the distribution model and use it to get the simulation of the slope in a given time interval with parameters chosen by users.

Appendix B

Detection Algorithm , Modeling And Simulation Code

B.1 Detection Algorithm

```
function [n,startpoint,duration,powerswing] = windrampdetect( N,A,
   alpha,beta,ps )
%function-windrampdetect
%N:number of all points in data,A:name of datasets as a matrix
%ps : Magnitude of change of wind power;
%alpha: minimum rate of ramp events
%beta: end rate, quals to the absolute value of ratio of current
   points power to largest power in the ramp events
%author:Xingbang Du
i = 1;
j = i +1;
%initial number
n = 0;
%lower bound of power wing
ps1 = -ps;
%record of all starting points of wind ramp events
startpoint = [];
%record of all duration of wind ramp events
duration = [];
%record of all power swing of wind ramp events
powerswing = [];
while 1
    j = i + 1;
    data = [];
```

```
for j = i+1:N
    %up wind ramp
    %rule 1,power swing must be larger than lower bound
    if (A(j,1)- A(i,1)> ps)
        %rule 2,rate of a ramp must be larger than alpha
        if (A(j,1)- A(i,1)>5*(j-i)*alpha)
            %find the maximum power from i point to j point

            pm = A(j,1);
            data = [data pm];

            %rule 3,check if wind is still in the ramp event
               after drop
            if A(j,1)>beta*max(data)
                continue;
            else
                sp = i;
                d = 5*(j - 1 - i);
                psw = A(j-1,1) - A(i,1);
                i = j - 1;
                n = n + 1;
                startpoint = [startpoint sp];
                duration = [duration d];
                powerswing = [powerswing psw];
                break;
            end
        else
            sp = i;
            d = 5*(j-1 - i);
            psw = A(j-1,1) - A(i,1);
            i = j - 1;
            n = n + 1;
            startpoint = [startpoint sp];
            duration = [duration d];
            powerswing = [powerswing psw];
            break;
        end
    %down wind ramp
    %rule 1
    elseif (A(j,1)- A(i,1)< ps1)
        %rule 2
        if (A(j,1)- A(i,1)<5*(i-j)*alpha)
            %find the minimum power from i point to j point

            pm = A(j,1);
            data = [data pm];
```

```
                mindata = min(data);
                %rule 3,check if wind power is still in the ramp
                   event after
                %raise
                if A(j,1)<(mindata/beta)
                    continue;
                else
                    sp = i;
                    d = 5*(j - 1 - i);
                    psw = A(j-1,1) - A(i,1);
                    i = j - 1;
                    n = n + 1;
                    startpoint = [startpoint sp];
                    duration = [duration d];
                    powerswing = [powerswing psw];
                    break;
                end
            else
                sp = i;
                d = 5*(j-1 - i);
                psw = A(j-1,1) - A(i,1);
                i = j - 1;
                n = n + 1;
                startpoint = [startpoint sp];
                duration = [duration d];
                powerswing = [powerswing psw];
                break;
            end
        else
            i = i + 1;
            break;
        end
    end
    %all points has been detected
    if i == N
        break;
    end

end


end
```

B.2 Modeling Code

```
function [tieslist] = NEWgettieslist(n,startpoint,duration,
   powerswing,DATAS)
```

```
%get time interval
tieslist = [];
endpointlist = [];
startplist = [];
for i =1:n
    endpoint = startpoint(i) + duration(i)/5;
    endpointlist = [endpointlist endpoint];
    endp = DATAS(endpoint);

end
for i =1:n
    sp1 = startpoint(i);
    startp = DATAS(sp1);
    startplist = [startplist startp];
end
for j =1:n
    if j+1>n
        break
    end
    ties = startpoint(j+1)-endpointlist(j);
    tieslist = [tieslist ties];
end
end

function sl = M1getslope(duration,powerswing,n)
%UNTITLED15
%
sl = [];
for i =1:n
    slp = powerswing(i)/duration(i);
    sl = [sl slp];
end
end

function mis = M1getmis(sl,n,alpha)
%find missing point
%for the down wind ramp events,if slope is samll than minus three
   times alpha
%it should be a missing point
mis = [];
for i = 1:n
    slt = - sl(i);
    if slt > 3*alpha
        mis = [mis i];
    end
end
end
```

Modeling of data sets of Spring 2014:

```
[n14,sp14,du14,powerswing14] = windrampdetect(26484,spring2014
   ,8.5,0.9,100 );
ti14 = NEWgettieslist(n14,sp14,du14,powerswing14,spring2014);
sl14 = M1getslope(du14,powerswing14,n14);
mis14 = M1getmis(sl14,n14,8.5);
nm14 = length(mis14);
n14m = n14 - nm14;
nof =  floor(n14m/4);
spplist = [];
for i=1:nm14
    ti14(i) = [];
    sl14(i) = [];
    du14(i) = [];
    sp14(i) = [];
end
for i=1:n14m
    sppower = spring2014(sp14(i));
    spplist = [spplist sppower];
end
sspplist = sort(spplist);
sspplist(nof)
sspplist(nof*2)
sspplist(nof*3)

sll1 = [];sll2 = [];sll3 = [];sll4 = [];
dul1 = [];dul2 = [];dul3 = [];dul4 = [];

for j = 1:n14m
    tpls = spring2014(sp14(j));
    if tpls < sspplist(nof)
        sll1 = [sll1 sl14(j)];
        dul1 = [dul1 du14(j)];
    end
    if tpls < sspplist(nof*2) && tpls >= sspplist(nof)
        sll2 = [sll2 sl14(j)];
        dul2 = [dul2 du14(j)];
    end
    if tpls < sspplist(nof*3) && tpls >= sspplist(nof*2)
        sll3 = [sll3 sl14(j)];
        dul3 = [dul3 du14(j)];
    end
    if tpls >= sspplist(nof*3)
        sll4 = [sll4 sl14(j)];
        dul4 = [dul4 du14(j)];
    end
end
```

```
nl1 = length(sll1);nl2 = length(sll2);nl3 = length(sll3);
nl4 = length(sll4);
sll1u = [];sll2u = [];sll3u = [];sll4u = [];
dul1u = [];dul2u = [];dul3u = [];dul4u = [];
sll1d = [];sll2d = [];sll3d = [];sll4d = [];
dul1d = [];dul2d = [];dul3d = [];dul4d = [];

for i=1:nl1
    if sll1(i)>0
        sll1u=[sll1u sll1(i)];
        dul1u=[dul1u dul1(i)];
    else
        sll1d=[sll1d -sll1(i)];
        dul1d=[dul1d dul1(i)];
    end
end
for i=1:nl2
    if sll2(i)>0
        sll2u=[sll2u sll2(i)];
        dul2u=[dul2u dul2(i)];
    else
        sll2d=[sll2d -sll2(i)];
        dul2d=[dul2d dul2(i)];
    end
end
for i=1:nl3
    if sll3(i)>0
        sll3u=[sll3u sll3(i)];
        dul3u=[dul3u dul3(i)];
    else
        sll3d=[sll3d -sll3(i)];
        dul3d=[dul3d dul3(i)];
    end
end
for i=1:nl4
    if sll4(i)>0
        sll4u=[sll4u sll4(i)];
        dul4u=[dul4u dul4(i)];
    else
        sll4d=[sll4d -sll4(i)];
        dul4d=[dul4d dul4(i)];
    end
end

n1u = length(sll1u);n2u = length(sll2u);n3u = length(sll3u);n4u =
   length(sll4u);
```

```
n1d = length(sll1d);n2d = length(sll2d);n3d = length(sll3d);n4d =
   length(sll4d);

dsl1u = reshape(sll1u,n1u,1);
dsl2u = reshape(sll2u,n2u,1);
dsl3u = reshape(sll3u,n3u,1);
dsl4u = reshape(sll4u,n4u,1);

ddu1u = reshape(dul1u,n1u,1);
ddu2u = reshape(dul2u,n2u,1);
ddu3u = reshape(dul3u,n3u,1);
ddu4u = reshape(dul4u,n4u,1);

dsl1d = reshape(sll1d,n1d,1);
dsl2d = reshape(sll2d,n2d,1);
dsl3d = reshape(sll3d,n3d,1);
dsl4d = reshape(sll4d,n4d,1);

ddu1d = reshape(dul1d,n1d,1);
ddu2d = reshape(dul2d,n2d,1);
ddu3d = reshape(dul3d,n3d,1);
ddu4d = reshape(dul4d,n4d,1);

%time interval
pdtii = reshape(ti14,length(ti14),1);
pdti = fitdist(pdtii,'gamma');

%slope
pds1u = fitdist(dsl1u,'exponential');
pds2u = fitdist(dsl2u,'exponential');
pds3u = fitdist(dsl3u,'exponential');
pds4u = fitdist(dsl4u,'exponential');

pds1d = fitdist(dsl1d,'exponential');
pds2d = fitdist(dsl2d,'exponential');
pds3d = fitdist(dsl3d,'exponential');
pds4d = fitdist(dsl4d,'exponential');
%duration
pdd1u = fitdist(ddu1u,'gamma');
pdd2u = fitdist(ddu2u,'gamma');
pdd3u = fitdist(ddu3u,'gamma');
pdd4u = fitdist(ddu4u,'gamma');

pdd1d = fitdist(ddu1d,'gamma');
pdd2d = fitdist(ddu2d,'gamma');
pdd3d = fitdist(ddu3d,'gamma');
pdd4d = fitdist(ddu4d,'gamma');
```

B.3 Simulation Code

```
x = 0;
y = 3354;
ps = 100;
giventime = 25000;
nps = 0;
dupredlist =[];
slpredlist =[];
tipredlist =[];
pspredlist =[];
xlist = [x];
ylist = [y];
yvlist= [0];
xline = [];
ps_splist = [];
while 1
    %get up or down
    if y < ps
      ppn = 1;
    end
    if y >= ps && y < sspplist(nof)
      ppn = length(sll1u)/length(sll1);
    end
    if y >= sspplist(nof) && y < sspplist(nof*2)
      ppn = length(sll2u)/length(sll2);
    end
    if y >= sspplist(nof*2) && y < sspplist(nof*3)
      ppn = length(sll3u)/length(sll3);
    end
    if y >= sspplist(nof*3) && y < sspplist(nof*4)
      ppn = length(sll4u)/length(sll4);
    end
    if y >= sspplist(nof*4)
      ppn = 0;
    end
    xpn = rand;
    %up events
    if xpn<ppn
        %get duration
        pddv = M4get_du_up(y,nof,sspplist,pdd1u,pdd2u,pdd3u,pdd4u);
        dupred = random(pddv)/5;
        dupredlist = [dupredlist dupred];
        %get end point
        xe = x + dupred;
        if xe > giventime
            break
```

```
        end
        xlist = [xlist x];
        xlist = [xlist xe];
        xlist = [xlist xe];
        ps_splist = [ps_splist x];
        %get slope
        pdsv = M4get_sl_up(y,nof,sspplist,pds1u,pds2u,pds3u,pds4u);
        slpred = random(pdsv);
        slpredlist = [slpredlist slpred];
        yvlist = [yvlist slpred];
        yvlist = [yvlist slpred];
        yvlist = [yvlist 0];
        %get power swing
        pspred = dupred*slpred;
        pspredlist = [pspredlist pspred];
        %updat y
        y = y + pspred;
    else
        %get duration
        pddd = M4get_du_down(y,nof,sspplist,pdd1d,pdd2d,pdd3d,pdd4d
           );
        dupred = random(pddd)/5;
        dupredlist = [dupredlist dupred];
        %get end point
        xe = x + dupred;
        if xe > giventime
            break
        end
        xlist = [xlist x];
        xlist = [xlist xe];
        xlist = [xlist xe];
        ps_splist = [ps_splist x];
        %get slope
        pdsd = M4get_sl_down(y,nof,sspplist,pds1d,pds2d,pds3d,pds4d
           );
        slpred = -random(pdsd);
        slpredlist = [slpredlist slpred];
        yvlist = [yvlist slpred];
        yvlist = [yvlist slpred];
        yvlist = [yvlist 0];
        %get power swing
        pspred = dupred*slpred;
        pspredlist = [pspredlist pspred];
        %updat y
        y = y + pspred;
    end
    nps = nps + 1;
```

```
    %get time interval
    tipred = random(pdti);
    tipredlist = [tipredlist tipred];
    x = xe + tipred;
    xlist = [xlist x];
    ps_splist = [ps_splist x];
    yvlist = [yvlist 0];
    if x > giventime
        break
    end
end
picps = plot(xlist,yvlist);
hold on
for i=1:length(xlist)
    xline = [xline 0];
end
picps = plot(xlist,xline);
```

Biography

Xingbang Du is a graduate student in the Department of Industrial & Systems Engineering of Lehigh University. Before he started graduate study, he got his bachelor degree in Nanjing University of Aeronautics and Astronautics, Nanjing, China.

In Lehigh University, under the guidance of the graduate thesis supervisor Prof. Defourny, Xingbang has finished the Simulation of prediction of Wind Ramp Events In Smart Grid part of the research.

In Nanjing University of Aeronautics and Astronautics, he has written a thesis: Large Aircraft Suppliers Coordinated Develop Network Optimization, optimize the density of the coordinated network of Large Aircraft Suppliers Coordinated Development in China.

EDUCATION

M.S., Lehigh University, 2018

B.S., Nanjing University of Aeronautics and Astronautics, 2016