AIAA Aviation
16-20 June 2014, Atlanta, GA
15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference

Multi-Fidelity Uncertainty Quantification: Application to a Vertical Axis Wind Turbine Under an Extreme Gust

A. Santiago Padrón*, Juan J. Alonso† and Francisco Palacios‡
Stanford University, Stanford, CA, 94305, USA

Matthew Barone§ and Michael S. Eldred¶
Sandia National Laboratories‖, Albuquerque, NM, 87185, USA

Designing better vertical axis wind turbines (VAWTs) requires considering the uncertain wind conditions they operate in and quantifying the effect of such uncertainties. We study the effect of an uncertain extreme gust on the maximum forces on the blades of the VAWT. The gust is parametrized by three random variables that control its location, length and amplitude. We propose a multi-fidelity approach to uncertainty quantification that uses polynomial chaos to create an approximation to the high-fidelity statistics via a correction function based on the difference between high and low-fidelity simulations. The multi-fidelity method provides accurate statistics on the maximum forces for a small number of simulations and the multi-fidelity statistics are consistent with the high-fidelity (CFD) statistics. We developed a practical method to simulate a gust, that changes its magnitude in the flow direction, in a CFD solver by combining the field velocity method (FVM) and the geometric conservation law (GCL). The ability to study the effect of the gust with the high-fidelity (CFD) solver is crucial as the low-fidelity (blade element/vortex lattice) solver underestimates the effect of the gust on the maximum forces.

I. Introduction

Quantifying the response of a wind turbine to an extreme wind gust is an important design requirement.[1] The aerodynamics of the gust interacting with the turbine blades are complex and difficult to model because of dynamic stall.[2] Different simulation tools can be used to model the interaction of the gust with the turbine blades. These models can be classified as "high-fidelity" or "low-fidelity". A high-fidelity model will accurately simulate the gust and blade interaction but at a high computational cost; whereas, a low-fidelity model will provide a less accurate simulation but at a low computational cost.

Low-fidelity tools are used extensively in the modeling of vertical axis wind turbines (VAWTs)[3,4,5] because of their low computational cost and relative good accuracy for normal operating conditions. However, their performance will be degraded in the presence of the gust.[2]

In current practice, only a single "most likely" extreme gust event is simulated. Whereas, in reality, the extreme gust is stochastic[6,7] and failure to account for the variability or uncertainty in the gust can lead to either over-conservative designs or, in some cases, under-design. Many simulations are needed to properly quantify the effect of the uncertain gust on the wind turbine. Using only a cheap low-fidelity model to perform the many simulations will likely not result in accurate statistics and relying exclusively on the high-fidelity model will give us accurate statistics but at a high cost.

*Ph.D. Candidate, Department of Aeronautics & Astronautics, padronas@stanford.edu, AIAA Student Member.
†Associate Professor, Department of Aeronautics & Astronautics, AIAA Associate Fellow.
‡Engineering Research Associate, Department of Aeronautics & Astronautics, AIAA Senior Member
§Principal Member of Technical Staff, Aerosciences Department, AIAA Senior Member.
¶Distinguished Member of Technical Staff, Optimization and Uncertainty Estimation Department, AIAA Associate Fellow.
‖Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Here we explore the use of multi-fidelity uncertainty quantification in order to reduce the computational cost of obtaining accurate statistics of the maximum forces on the wind turbine blades by combining a small number of high fidelity simulations with a large number of low-fidelity simulations (section III.B.3). The multi-fidelity uncertainty quantification approach is similar to previous work in Ref. 8, but here we apply it to a more complex problem and for which the low-fidelity model is not a very good replacement of the high-fidelity model.

In this work, we consider a computational fluid dynamics (CFD) solver as the high-fidelity tool and a blade element/vortex lattice aerodynamic model as the low-fidelity tool. To the best of our knowledge we perform the first practical CFD simulation of an extreme wind gust with a wind turbine. This is done by combining the Field Velocity Method (FVM) with the Geometric Conservation Law (GCL) (section III.A.3).

The uncertain gust and the vertical axis wind turbine considered in this problem are described in section II. The the high and low-fidelity simulation tools are described in section III.A. The uncertainty quantification method is described in section III.B and the results follow in section IV

II. Problem definition: The uncertain gust and the VAWT

For an extreme gust, the shape, size, and position are uncertain. Here we take the gust shape as fixed and treat the variables that determine the size and position of the gust in a probabilistic setting to properly determine the output distribution and statistics for the maximum loads. The extreme gust shape we use is the one specified in the IEC standard[1]

$$u_g(x) = \begin{cases} U_\infty - 0.37u_e \sin\left(\frac{3\pi(x-x_0)}{L_g}\right)\left[1-\cos\left(\frac{2\pi(x-x_0)}{L_g}\right)\right], & \text{if } 0 \le x - x_0 \le L_g; \\ U_\infty, & \text{otherwise;} \end{cases} \tag{1}$$

where U_∞ is the free-stream velocity, x_0 is the gust starting position, L_g is the gust length, and u_e is the gust amplitude. An example of this gust is shown in fig. 1.

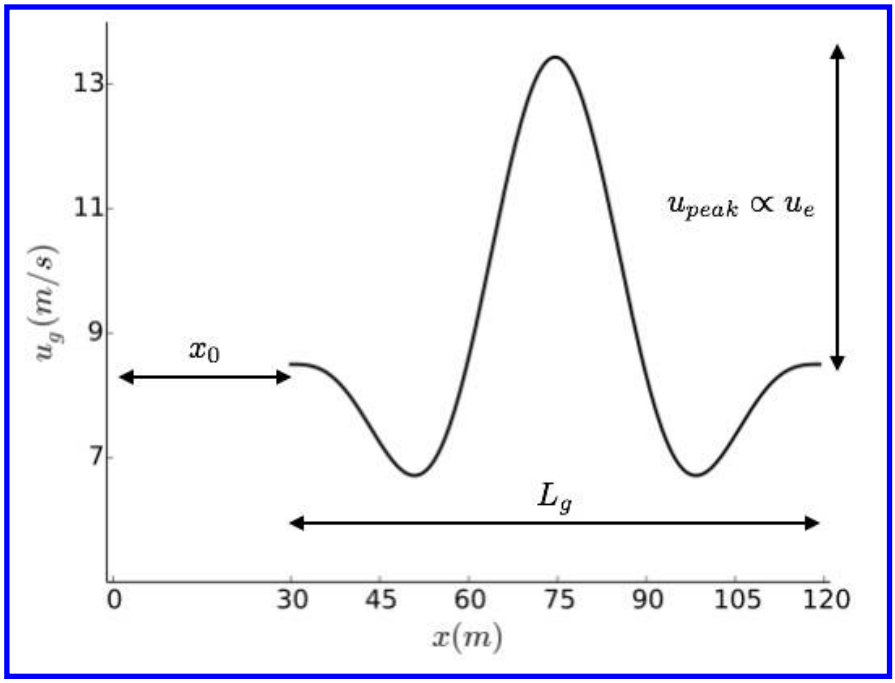


Figure 1. Example of an extreme gust with gust parameters (x_0, L_g, u_e).

II.A. Uncertain variables

For the fixed gust shape given by eq. (1), we treat its parameters (x_0, L_g, u_e) as uncertain variables that follow the probability distributions listed in table 1. For the particular turbine considered in this work

(section II.E and table 2), the numerical values of the variables of the distributions are also listed in table 1, and the corresponding probability distribution functions are shown in fig. 2.

II.B. Gust starting position x_0

The gust starting position x_0 follows a uniform distribution in the interval $[-246.86\,\text{m}, -179\,\text{m}]$ that accounts for all possible interactions of the gust with the turbine blades. The $x = 0\,\text{m}$ position is at the turbine center of rotation.

II.C. Gust length L_g

The gust length L_g follows a normal distribution with mean $\mu_{L_g} = U_\infty T_g$, where the gust time scale ($T_g = 10.5\,\text{s}$) is the deterministic value given in the IEC standard. For the standard deviation, we picked a reasonable estimate of 10% of the mean value, $\sigma_{L_g} = 0.1\,\mu_{L_g}$.

Uncertain parameter	Probability distribution	Distribution variables
x_0	Uniform (a, b) $f(x_0) = \frac{1}{b-a}$	$a = -246.86\,\text{m}$, $b = -179\,\text{m}$
L_g	Normal (μ, σ) $f(L_g) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{L_g-\mu}{\sigma})^2}$	$\mu = 89.25\,\text{m}$, $\sigma = 8.925\,\text{m}$
u_e	Gumbel (α, β) $f(\zeta)^\dagger = \alpha \exp[-e^{-\alpha(\zeta-\beta)}] e^{-\alpha(\zeta-\beta)}$	$\alpha = 1.14$, $\beta = 11.24$

Table 1. Uncertain parameters for the extreme gust and their probability distributions. The values of the distribution variables are for the particular turbine considered in this work (see table 2).

† The gust amplitude is obtained from $u_e = \sigma_u \zeta$.

II.D. Gust amplitude u_e

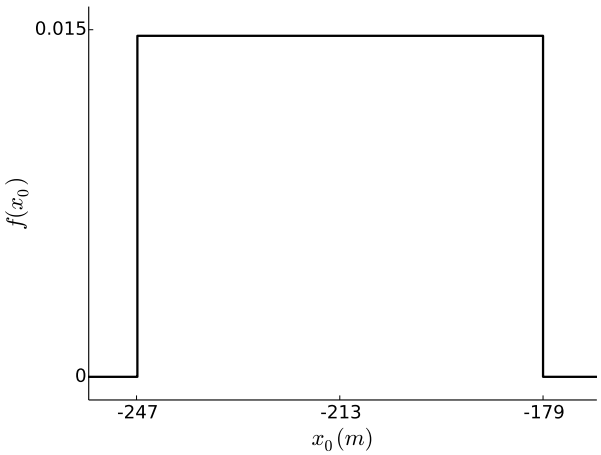
The gust amplitude u_e follows a gumbel distribution. This gumbel distribution can be derived from the wind climate given in the IEC standard as shown in detail in Larsen and Hansen.[7] Here we briefly describe how to obtain the parameters (α, β) that determine the Gumbel distribution,

$$f(\zeta) = \alpha \exp[-e^{-\alpha(\zeta-\beta)}] e^{-\alpha(\zeta-\beta)}, \quad \alpha = \frac{1}{2C(z)}, \quad \beta = 2C(z) \ln(T\kappa), \tag{2}$$

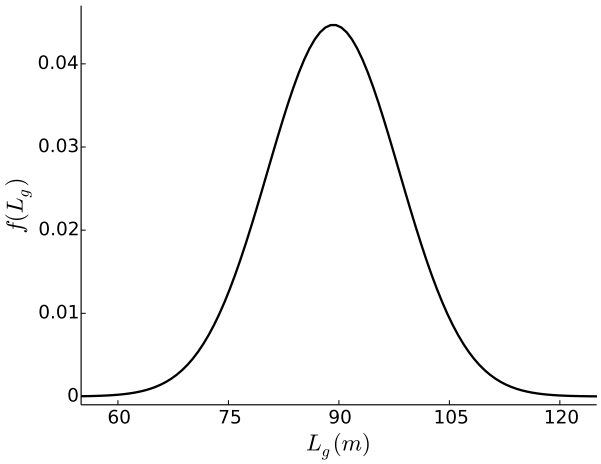
where $\zeta = {}^{u_e}/_{\sigma_u}$ is the gust amplitude normalized by the standard deviation of the wind velocity, and T is the recurrence period for an extreme gust. We set $T = 50$ years, which is a common value in wind turbine design. Expressions for the standard deviation of the wind velocity σ_u, the terrain and altitude dependent empirical constant $C(z)$, and the expected rate of local extremes κ, can be found in Larsen and Hansen.[7] These expressions depend on the particular geometry and operating conditions of the turbine. Using the turbine considered in this work (section II.E and table 2) we obtain the distribution variables listed in table 1 and the probability distribution for ζ and then u_e ($u_e = \sigma_u \zeta$) that is shown in fig. 2(c).

II.E. VAWT description

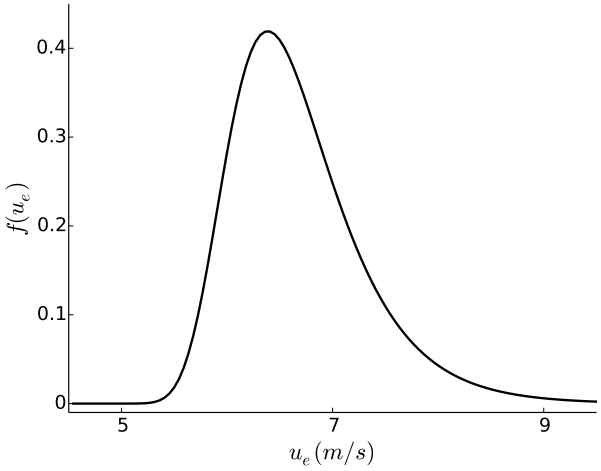
The wind turbine considered is a preliminary design generated at Sandia National Laboratories for a 3-bladed, 5 MW, offshore vertical-axis wind turbine (VAWT) (fig. 3(a)). The maximum rotor radius is 54 meters, and the rotor height measured from the bottom blade attachment point to the blade tips is 104.7 meters. A central tower extends from the rotor base to 70% of the rotor height, and is attached to the blades via three support struts. The present analysis considers a two-dimensional horizontal slice through the rotor at the



(a) Uniform distribution for the gust starting position.



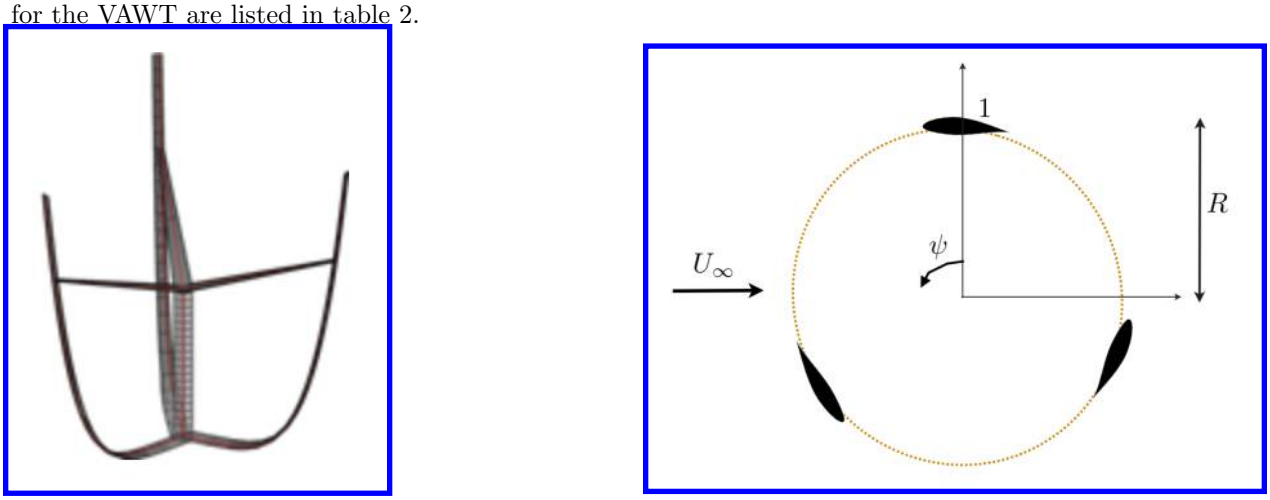
(b) Normal distribution for the gust length.



(c) Gumbel distribution for the gust amplitude.

Figure 2. Probability distributions for the uncertain variables.

blade tips (fig. 3(b)). This is the region of most interest from the aerodynamics perspective, given that this is the location where the maximum forces are generated per unit span. The blade chord at this location is 2 meters, with the blade attachment point located at 25% of the chord (measured from the leading edge). The airfoil section is a SNL 0018/50, a natural laminar flow airfoil developed specifically for VAWTs. The design RPM at rated power for this turbine is 7.4 RPM. A summary of the geometry and the operating conditions for the VAWT are listed in table 2.



(a) The 5 MW VAWT.

(b) A schematic of a 2D horizontal slice of the VAWT.

Figure 3. The vertical axis wind turbine and a schematic for the two-dimensional horizontal slice through the tips of the VAWT considered in this work.

III. Methods: Modeling the gust and uncertainty quantification

In this section we describe the two important ideas put forth in this paper 1) a practical method to simulate a wind gust for wind turbine applications in a CFD solver and 2) a multi-fidelity approach to uncertainty quantification. Also, we describe the low and high-fidelity tools used to simulate the wind turbine and we describe polynomial chaos, the uncertainty quantification method the proposed multi-fidelity approach uses.

III.A. Modeling of the gust interaction with the VAWT

The are many aerodynamic models available to simulate a wind turbine,[3,4,5,9,10,11] ranging from simple analytic models all the way to Direct Numerical Simulation (DNS), where each model depending on the context can be considered a high-fidelity or low-fidelity model. For this paper we consider the Euler equations in SU^2 (a CFD solver) as the high-fidelity model and CACTUS, a blade element/vortex lattice aerodynamic solver, as the low-fidelity model. The difference in simulation time between SU^2 and CACTUS (1 simulation takes 1 min on 1 processor) is around 3 orders of magnitude. These models are described below as well as how to model a gust in SU^2. The modeling of the gust in a CFD solver (SU^2) is not a straight forward task for this reason a subsection is devoted to introduce a practical method to model the gust. On the other hand modeling the gust in a low-fidelity tool such as CACTUS is straightforward, all that is needed is to modify the free-stream velocity as a function of time.

III.A.1. SU^2- High-fidelity model

The high-fidelity model is Stanford University Unstructured (SU^2).[12] SU^2 is a C++ software suite developed for the specific task of solving partial differential equations (PDE) and PDE-constrained optimization problems on general unstructured meshes. The computational fluids dynamics (CFD) solver in SU^2 suite has been used and validated for many aerospace engineering applications.[13]

In this work we use the compressible two-dimensional Euler equations to model the aerodynamics of the gust interacting with the vertical axis wind turbine. We use the Euler solver in SU^2 as opposed to the RANS solver, because of the cheaper computational cost without any significant loss of accuracy for our problem.

Turbine data	Specification
Operating data	
Rated capacity (MW)	5
Terrain type	Offshore
Average wind speed U_∞ (m/s)	8.5
Reynolds number	5.4×10^6
Tip speed ratio λ	5
Geometry data	
Number of rotor blades	3
Rotor diameter (m)	108
Blade chord c (m)	2.00
Blade attachment point	$0.25c$
Height from base of rotor (m)	104.7
Airfoil	SNL 0018/50
Specific data to calculate u_e (section II.D)	
Turbulence intensity I_{ref}	0.16
Reference wind speed U_{ref} (m/s)	37.5
Mean wind speed bin width (m/s)	2
Extreme gust recurrence period T (year)	50

Table 2. Vertical axis wind turbine description.

The Euler equations are solved on a domain $\Omega \subset \mathbb{R}^2$, with airfoil boundaries S and far-field boundary Γ_∞,

$$\begin{aligned} \partial_t U + \partial_x F + \partial_y G &= 0, && \text{in } \Omega, \\ (\vec{u} - \vec{x}_t) \cdot \vec{n}_S &= 0, && \text{on } S, \\ W_+ &= W_\infty, && \text{on } \Gamma_\infty, \end{aligned} \tag{3}$$

where

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \; F = \begin{bmatrix} \rho(u - x_t) \\ \rho u(u - x_t) + p \\ \rho v(u - x_t) \\ \rho E(u - x_t) + pu \end{bmatrix}, \; G = \begin{bmatrix} \rho(v - y_t) \\ \rho u(v - y_t) \\ \rho v(v - y_t) + p \\ \rho E(v - y_t) + pv \end{bmatrix},$$

ρ is the fluid density, $\vec{u} = [u, v]^{\mathrm{T}}$ is the fluid velocity in a cartesian coordinate system, $\vec{x}_t = [x_t, y_t]^{\mathrm{T}}$ is the boundary velocity (mesh velocitiy after discretization), E is the total energy per unit mass, and p is the static pressure

$$p = (\gamma - 1)\rho \left(E - \frac{1}{2}(u^2 + v^2) \right).$$

The second line of eq. (3) represents the flow tangency condition at a solid wall, and the third line represents a characteristic-based boundary condition at the far-field where the fluid state at the boundary is updated using the state at infinity depending on the sign of the eigenvalues.

In the numerical implementation, the Euler equations eq. (3) are discretized using a standard edge-based finite volume formulation on the dual grid, obtained by applying the integral formulation of the equations to a dual grid control volume surrounding any given node of the grid and performing an exact integration around the outer boundary of this control volume. The convective flux is discretized using a central scheme with JST-type artificial dissipation.[14] Time integration is implicit and handled by a second-order accurate dual-time stepping approach.[15]

III.A.2. CACTUS - Low-fidelity model

The low-fidelity model is CACTUS, the Code for Axial and Crossflow Turbine Simulation developed at Sandia National Laboratories (SNL).[16] CACTUS simulates the performance and loads of a wind turbine rotor in the time domain using a blade element/vortex lattice aerodynamic model. The rotor blades are discretized into blade element sections, where each section is assigned two-dimensional airfoil performance characteristics in the form of tables containing lift, drag, and moment coefficient vs. angle of attack. CACTUS also incorporates unsteady aerodynamic blade load effects such as dynamic stall, apparent mass and blade rotation effects. For the wake flow field, CACTUS uses a potential flow model comprised of free vortex line elements shed and trailed from each blade element at each time step.

III.A.3. Gust modeling in a CFD solver (SU^2)

Modeling a wind gust in a CFD solver is non-trivial. Initializing the gust in the flow field or specifying the gust as a time-dependent boundary condition (this approach is used for the low-fidelity tool) will not work. The reason for this is that the specified gust should satisfy the Euler equations eq. (3) or at least the gust velocity field $\vec{V_g}$ should be divergence free $\nabla \cdot \vec{V_g} = 0$. The extreme gust, specified in the IEC standard, and considered here

$$\vec{V_g} = \begin{bmatrix} u_g(x) = eq.\ (1) \\ v_g = 0 \end{bmatrix}, \tag{4}$$

changes its magnitude in the free-stream direction and has non-zero divergence

$$\nabla \cdot \vec{V_g} = \partial_x u_g + \partial_y v_g = \partial_x u_g(x) \neq 0. \tag{5}$$

In the rest of this section we describe a practical method that will allow us to simulate a gust that changes its magnitude in the free-stream direction and whose velocity field has non-zero divergence.

Even though the wind turbine gust has non-zero divergence, there are divergence free gusts in aerospace applications and efficient methods to simulated them have been developed. Two examples are a gust whose magnitude changes are perpendicular to the free-stream which is used for gust alleviation studies in aircraft and a vortex gust used to study blade vortex interactions that occur in rotorcraft. An efficient method used to study the effect of these divergence free gusts is the Field Velocity Method (FVM).[17,18,19,20,21] This method prescribes the gust as grid velocities which alleviates the problem of needing small cells everywhere in the mesh in order to avoid the gust being dissipated. For a CFD code that already has the capability to simulate moving meshes all that is need is to specify the gust as the negative of the grid velocity in the Euler equations eq. (3), $x_t = -u_g$. Our case, has true grid velocities due to the rotation of the wind turbine, so for cases with true grid velocities $x_{t_{true}}$ we would set $x_t = x_{t_{true}} - u_g$.

Note that the FVM is not quite equivalent to the Euler equations because of the extra prescribed gust velocity. Also, as the gust velocity is prescribed only the effect of the gust on the body is captured, the effect of the body on the gust is not. As long as the body does not affect the source of the gust which is almost always the case and which is the case here, the FVM method works remarkably well.[17,18,19,20,21]

Directly applying the FVM to the non-divergence free extreme gust causes non-physical behavior and the method fails terribly. Using the recently developed Split Velocity Method (SVM)[22] an extension to the FVM that captures the effect of the body on the gust by introducing additional source terms in eq. (3) also does not work to simulate the gust.

When performing simulation with dynamic meshes (grid velocities) it has been shown that Geometric Conservation Law (GCL)[23,24,25] should be satisfied for consistency in the governing equations. Usually adding GCL provides a slight improvement in accuracy.

Here we found that satisfying the GCL is crucial when using the FVM method, as this combination allows us to successfully simulate the extreme gust and in fact any non-divergence free gust. To our surprise combining the GCL with the SVM does not work.

In summary, a practical method to model a gust that changes magnitude in the free-stream direction (a non divergence free gust) requires the use of the field velocity method in combination with the geometric conservation law.

III.B. A multi-fidelity approach to propagating the gust uncertainty

Uncertainty quantification (UQ) is the process of 1) characterizing input uncertainties, 2) propagating these uncertainties through a computational model, and 3) quantifying the effect of the input uncertainties on the output(s) of interest. In essence, uncertainty quantification aims to gain a quantitative understanding of how variations in the inputs of a model affect the outputs. For instance, how the uncertain gust parameters affect the maximum load on the turbine. The first step of characterizing the input uncertainties was done in section II and the last step of characterizing the output of interest is done in the results section (section IV). In this section, we describe a novel approach to the second step of propagating the uncertainties by using computational models of different fidelities.

Different methods for performing the propagation of the input uncertainties depend on the type of the input uncertainties, either epistemic or aleatory. Epistemic uncertainties are a result of our lack of knowledge. Aleatory uncertainties, which are the ones we consider in this paper, are a result of their natural inherent variability, such as the wind speed. Because aleatory uncertainties are variable, they are analyzed with probabilistic methods and described by probability distributions.

A common way to study the effect of a probabilistic input on the output of a model is by Monte Carlo simulations (MC) or similar sampling methods, but one of their drawbacks is that many simulations of the model are needed to obtain accurate statistics of the outputs. Another approach is to use a stochastic expansion method, such as polynomial chaos (PC). In polynomial chaos, to approximate the statistics of the outputs a polynomial function is constructed that maps the uncertain inputs to the outputs of interest.

In this work, we use polynomial chaos, as opposed to Monte Carlo, because the functional relationship provided by polynomial chaos can be used to combine different levels of simulation fidelity (see section III.B.3) and accurate statistics on the output can be obtained in many fewer simulations using polynomial chaos for problems with few uncertain variables, which is the case in this study.

III.B.1. Description of the 1D polynomial chaos method

Polynomial chaos is a method of propagating the uncertainties in a model's inputs to the model's outputs, by constructing a polynomial approximation to the model's response to stochastic input parameters ξ,

$$R(\xi) \approx \hat{R}(\xi) = \sum_{i=0}^{P} \alpha_i \phi_i(\xi), \tag{6}$$

where the larger the polynomial order P the closer the approximation $\hat{R}(\xi)$ is to the true response $R(\xi)$. The basis functions $\phi_i(\xi)$ are known orthogonal polynomials and the coefficients α_i are what need to be estimated in the polynomial chaos method. One approach to calculate the PC coefficients α_i is the spectral projection method which takes advantage of the orthogonality of the basis polynomials. In the spectral projection method the first step to calculate the chaos coefficients is to multiply the PC expansion eq. (6) by $\phi_j(\xi)$ and $\rho(\xi)$, the density of the stochastic parameter, and then integrate over the probability space Ω,

$$\int_\Omega R(\xi)\phi_j(\xi)\rho(\xi)d\xi = \int_\Omega \sum_{i=0}^{\infty} \alpha_i \phi_i(\xi)\phi_j(\xi)\rho(\xi)d\xi. \tag{7}$$

Using the orthogonality of the basis polynomials with respect to the uncertain variable density,

$$\int \phi_i(\xi)\phi_j(\xi)\rho(\xi)d\xi = \delta_{i,j}, \tag{8}$$

and solving for the coefficients from eq. (7), we obtain

$$\alpha_i = \frac{\int_\Omega R(\xi)\phi_i(\xi)\rho(\xi)d\xi}{\int_\Omega \phi_i^2(\xi)\rho(\xi)d\xi} = \frac{\langle R(\xi), \phi_i(\xi)\rangle}{\langle \phi_i^2(\xi)\rangle} = \frac{1}{\langle \phi_i^2(\xi)\rangle}\int_\Omega R(\xi)\phi_i(\xi)\rho(\xi)d\xi. \tag{9}$$

A couple notes to keep in mind when evaluating the polynomial chaos coefficients equation eq. (9):

- The polynomial basis $\phi_i(\xi)$ is chosen such that it is orthogonal to the probability density function of the stochastic parameters $\rho(\xi)$ up to a constant. The common probability distributions have known basis polynomials,[26] For stochastic parameters that do not follow one of the common probability distributions, a numerically generated polynomial basis can be constructed.[27,28] The polynomial basis for the uncertain parameters considered in this paper are listed in table 3.

- The majority of the effort in solving for the coefficients resides in evaluating the integral, $\int_\Omega R(\xi)\phi_i(\xi)\rho(\xi)d\xi$, especially when $R(\xi)$ is expensive to evaluate, as the inner product $\langle\phi_i^2(\xi)\rangle$ for a particular polynomial basis is known. A common approach to evaluate the integral is by numerical quadrature

$$\sum_{j=0}^{N}\omega_j R(\xi_j)\phi_i(\xi_j)\rho(\xi_j), \tag{10}$$

  where the weights ω_j and quadrature points ξ_j depend on the quadrature method used.

Once the coefficients α_i are known on top of having a functional relationship between the uncertain inputs and outputs

$$\hat{R}(\xi) = \sum_{i=0}^{P}\alpha_i\phi_i(\xi), \tag{11}$$

estimates of the statistics for the mean and variance can be computed from

$$\mu_R = \alpha_0, \tag{12}$$

$$\sigma_R^2 = \sum_{i=1}^{P}\alpha_i^2\langle\phi_i^2(\xi)\rangle. \tag{13}$$

Uncertain parameter	Probability distribution	Polynomial basis
x_0	Uniform	Legendre
L_g	Normal	Hermite
u_e	Gumbel	Numerically generated

Table 3. The polynomial basis used in the polynomial chaos method for each uncertain parameter.

III.B.2. Multi-dimensional polynomial chaos and sparse grid construction

For multiple uncertain variables $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots \xi_d)$ the polynomial chaos expansion eq. (6) is written as

$$R(\boldsymbol{\xi}) \approx \hat{R}(\boldsymbol{\xi}) = \sum_{\mathbf{i}\in\mathcal{I}_p}\alpha_{\mathbf{i}}\boldsymbol{\Phi}_{\mathbf{i}}(\boldsymbol{\xi}), \tag{14}$$

where $\mathbf{i} = (i_1, i_2, \ldots, i_d)$ is a multi-index and the values of the elements $i_j \in \mathbb{N}$ depend on how the expansion is truncated, i.e., on how the index set $\mathcal{I}_p$ is defined. There are two common ways in which to define the index set: "total-order expansion" and "tensor-product expansion". In "total-order expansion" a total polynomial order bound p is enforced

$$\mathcal{I}_p = \{\mathbf{i} : |\mathbf{i}| \le p\}, \quad |\mathbf{i}| = i_1 + i_2 + \cdots + i_d. \tag{15}$$

Whereas in "tensor-product expansion" a per-dimension polynomial order bound p_j is enforced

$$\mathcal{I}_p = \{\mathbf{i} : i_j \le p_j,\ j = 1, \ldots, d\}. \tag{16}$$

The basis functions $\boldsymbol{\Phi}_{\mathbf{i}}(\boldsymbol{\xi})$ are given by products of one-dimensional orthogonal polynomials ϕ_{i_j},

$$\boldsymbol{\Phi}_{\mathbf{i}}(\boldsymbol{\xi}) = \prod_{j=1}^{d}\phi_{i_j}(\xi_j). \tag{17}$$

Similarly to the 1D case, the coefficients of the expansion are calculated from

$$\alpha_{\mathbf{i}} = \frac{1}{\langle\boldsymbol{\Phi}_{\mathbf{i}}^2(\boldsymbol{\xi})\rangle}\int_\Omega R(\boldsymbol{\xi})\boldsymbol{\Phi}_{\mathbf{i}}(\boldsymbol{\xi})\rho(\boldsymbol{\xi})\mathrm{d}\boldsymbol{\xi}, \tag{18}$$

where $\rho(\boldsymbol{\xi}) = \prod_{j=1}^{d} \rho_j(\xi_j)$ is the joint probability density of the stochastic parameters over the support $\Omega = \Omega_1 \times \cdots \times \Omega_d$. Same as in the 1D case the majority of the effort in computing the coefficients resides in evaluating the integral $\int_\Omega R(\boldsymbol{\xi})\boldsymbol{\Phi}_\mathbf{i}(\boldsymbol{\xi})\rho(\boldsymbol{\xi})\mathrm{d}\boldsymbol{\xi}$. To evaluate the integral tensor product quadrature $\mathcal{Q}_\mathbf{i}$ can be used, but the number of quadrature points in tensor product quadrature grows exponentially with the number of dimensions d. To alleviate the exponential growth in quadrature points sparse grid quadrature can be used. Sparse grid quadrature is constructed from a linear combination of tensor product quadrature grids $\mathcal{Q}_\mathbf{i}$ in such a way that high accuracy is preserved while using only a relative small number of grid(quadrature) points.[8,29,30] The isotropic sparse grid level q ($q \in \mathbb{N}$) used to perform the integration in eq. (18) is defined as

$$\mathcal{A}_{q,d}(R) = \sum_{q-d+1 \leq |\mathbf{i}| \leq q} (-1)^{q-|\mathbf{i}|} \binom{d-1}{q-|\mathbf{i}|} \mathcal{Q}_\mathbf{i}(R) \tag{19}$$

where as q increases more quadrature points are used and the integral is evaluated more accurately. The number of quadrature points depends on the number of uncertain variables and on the distributions of the uncertain variables. Table 4 lists the number of quadrature points used for our particular problem where we have $d = 3$ uncertain variables which follow the distributions in table 3.

III.B.3. Multi-fidelity uncertainty quantification with polynomial chaos

The goal of the multi-fidelity uncertainty quantification is to accurately characterize the high-fidelity system response $R_{high}(\boldsymbol{\xi})$ with less computational effort by using information obtained from the low-fidelity system response $R_{low}(\boldsymbol{\xi})$ as opposed to solely using the high-fidelity response. There are different methods of incorporating the information from the low-fidelity response.[8] Here we will do so by means of an additive correction function,

$$R_{corr}(\boldsymbol{\xi}) = R_{high}(\boldsymbol{\xi}) - R_{low}(\boldsymbol{\xi}). \tag{20}$$

In multi-fidelity UQ instead of approximating the high-fidelity function directly via polynomial chaos

$$R_{high}(\boldsymbol{\xi}) \approx \hat{R}_{high}(\boldsymbol{\xi}) = \sum_{\mathbf{i} \in \mathcal{I}_p} \alpha_\mathbf{i} \boldsymbol{\Phi}_\mathbf{i}(\boldsymbol{\xi}), \tag{21}$$

we approximate it from the polynomial chaos expansions of the low-fidelity and correction function

$$R_{high}(\boldsymbol{\xi}) \approx \tilde{R}_{high}(\boldsymbol{\xi}) = \hat{R}_{low}(\boldsymbol{\xi}) + \hat{R}_{corr}(\boldsymbol{\xi}). \tag{22}$$

The multi-fidelity approach could be beneficial if an accurate expansion of the correction function $\hat{R}_{corr}$ can be obtained with less computational effort, i.e., with less simulations(quadrature points) than the high-fidelity function $\hat{R}_{high}$ and if the low-fidelity model is a good approximation to the high-fidelity model.[8] A priori, it is hard to know for a particular problem if the conditions on the low-fidelity and correction function would be satisfied and even then if the multi-fidelity approach will lead to any computational savings. But for some problems large (80%) computational savings can be obtained.[8]

Sparse grid level	Number of evaluations
1	9
2	44
3	158
4	455
5	1099

Table 4. Number of function evaluations (quadrature points) in each sparse grid level. The higher levels reuse the evaluations in the lower levels.

IV. Results

First we show results for a particular gust realization and then we follow with the results from the uncertainty quantification study.

A simulation in SU^2 of the gust convecting passed the turbine blades is shown in fig. 4. To help visualization a circle is drawn around the blade we will refer to as "*Blade*-1". The position of *Blade*-1 in figs 4(a) and 3(b) is the 0° position. *Blade*-1 advances by 90° in each frame and each row of fig. 4 represents a full revolution of the turbine.

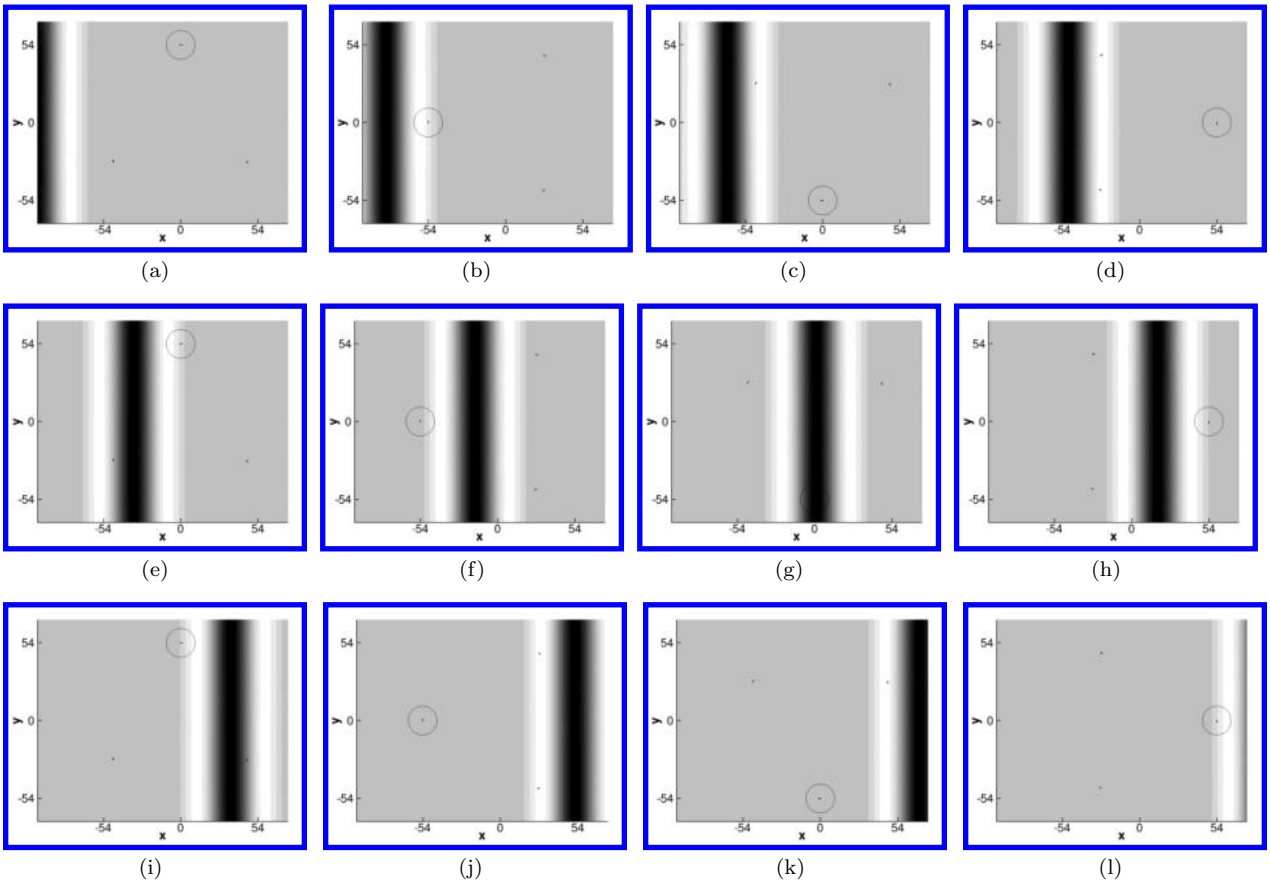


Figure 4. Evolution of the gust. A circle is drawn around *Blade*-1 to help with visualization. In each frame *Blade*-1 advances by 90° and each row represents a full revolution of the blade. The gust interactions lasts from 3 to 4 revolutions depending on the uncertain gust realization. The figure shows the gust velocity as compared to the free-stream velocity and the darker the color the higher the gust value.

As the gust convects pass the blades it changes the blades angles of attack. The angle of attack that a blade sees during a revolution for normal operating conditions (no-gust) is shown in fig. 5. If the gust interacts with the blade when it is already at a high angle of attack it can cause the flow to separate and the blade to stall. An angle of attack sweep for the blade (SNL 0018/50 airfoil) showed that the blade stalls at around 12° for both Euler and RANS simulations in SU^2. Also, the Euler and RANS simulations agreed well given us confidence that the results obtain by the Euler simulations are reasonable even if gust caused the blade to stall.

The most interaction *Blade*-1 has with the gust is during the second revolution of the turbine (figs. 4(e), 4(f), 4(g) and 4(h)). The normal and tangential forces on *Blade*-1 during the second revolution are shown in figs. 6 and 7, respectively. For comparison purposes (figs. 6 and 7) include the forces obtained from the low-fidelity model and for the normal operating conditions when there is no gust. The low-fidelity model under predicts the maximum forces when there is no gust and more noticeably when there is a gust. The gust has a bigger effect on the high-fidelity simulation because on top of changing the angle of attack the gust can cause unsteady aerodynamic effects which are not captured by the low-fidelity model. For instance the gust can cause the blade to stall, followed by a vortex convecting over the blade which can cause large variations in the forces and the larger discrepancies between the high and low-fidelity model.

For the uncertainty quantification study, we picked the maximum normal $C_{n,max}$ and maximum tangential force $C_{t,max}$ as our quantities of interest (QOI). We picked these QOI because as seen in figs. 6 and 7 they are affected by the uncertain gust and if they are large enough they can cause turbine failure. The gust

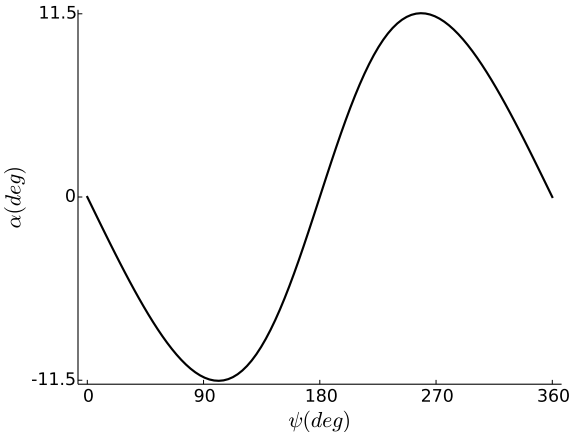
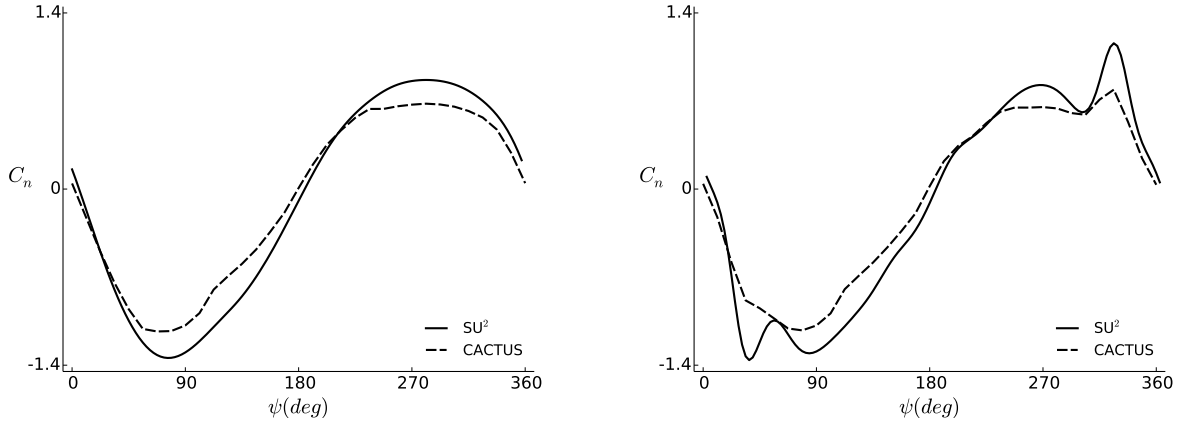
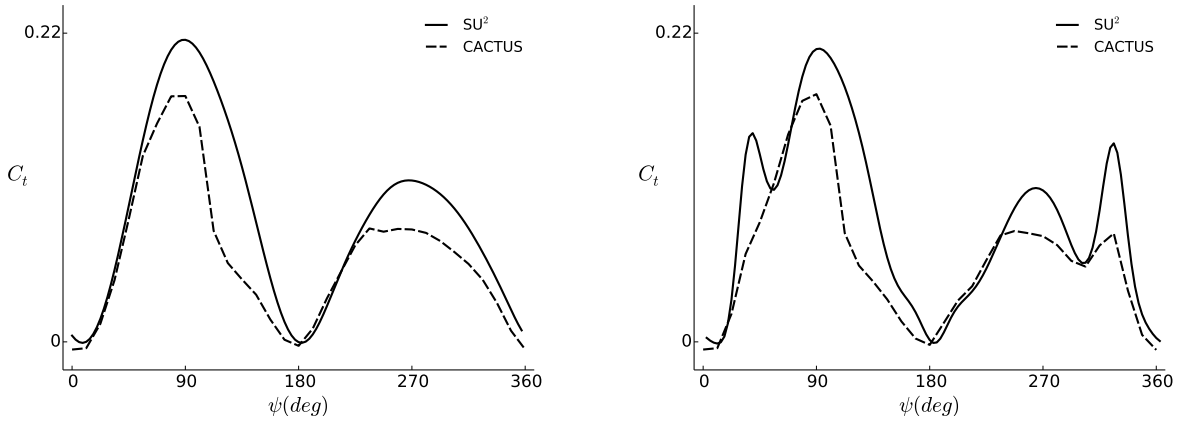


Figure 5. Angle of attack seen by the blades in normal operating conditions.



(a) No Gust (b) Gust

Figure 6. Comparison of the normal force coefficient C_n as a function of azimuth ψ between the high-fidelity tool SU^2 and the low-fidelity tool CACTUS. For the normal operating conditions CACTUS compares well with SU^2 (a), but CACTUS fails to capture all the variability introduced by the gust (b), which is expected of the low-fidelity tool.

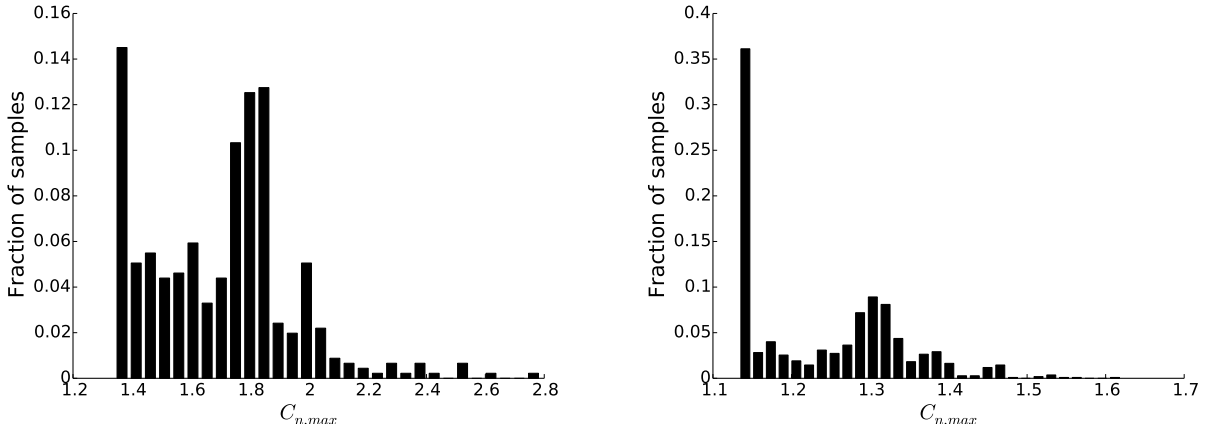


(a) No Gust (b) Gust

Figure 7. Comparison of the tangential force coefficient C_t as a function of azimuth ψ between the high-fidelity tool SU^2 and the low-fidelity tool CACTUS. For the normal operating conditions CACTUS compares well with SU^2 (a), but CACTUS fails to capture all the variability introduced by the gust (b), which is expected of the low-fidelity tool.

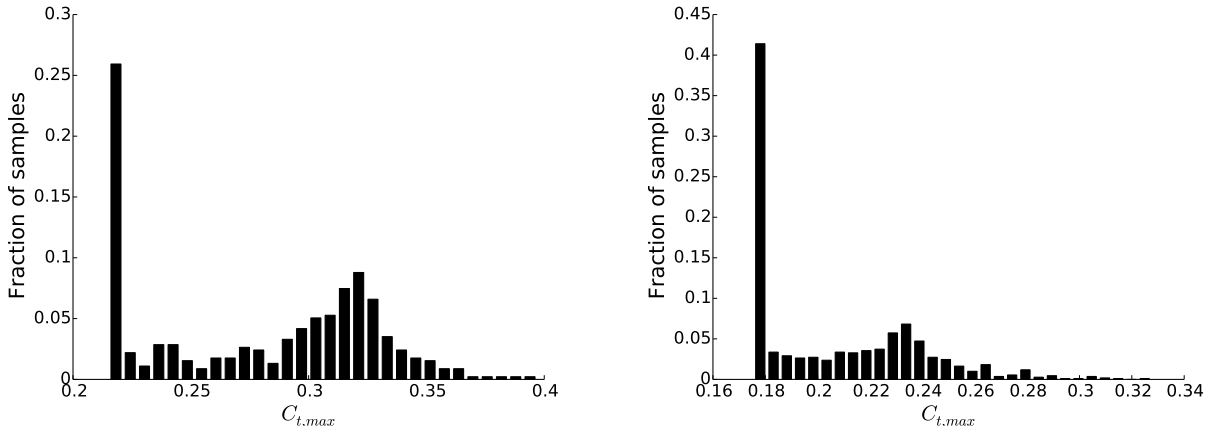
interaction in figs. 6 and 7 is a very benign interaction as the effect on the maximum forces is small.

The maximum number of gust realizations used for the high-fidelity case is 455 (sparse grid level 4) and for the low-fidelity case is 1099 (sparse grid level 5). A high-fidelity simulation takes about 30 minutes on 36 processors and a low-fidelity simulation 1 minute on 1 processor. The difference in simulation time between the high and low-fidelity is 3 orders of magnitude. Table 4 lists the number of samples in each sparse grid level. The higher sparse grid levels reuse the simulations form the lower sparse grid levels. A histogram of the output for the maximum normal force for the low and high-fidelity tool is shown in fig. 8 and for the maximum tangential force in fig. 9. There are many cases for which the gust does not affect the maximum forces, this is especially the case for the low-fidelity tool for which about 40% of its samples have their normal operating condition maximum. The spread in the high-fidelity is more pronounced and the maximum force can reach very large values as compared to the low-fidelity results.



(a) High-fidelity - $C_{n,max}$ (b) Low-fidelity - $C_{n,max}$

Figure 8. Histogram of the maximum normal force for the high-fidelity (455 samples) and low-fidelity (1099 samples).



(a) High-fidelity - $C_{t,max}$ (b) Low-fidelity - $C_{t,max}$

Figure 9. Histogram of the maximum tangential force for the high-fidelity (455 samples) and low-fidelity (1099 samples).

The sparse grid level 4 samples are shown in fig. 10. These samples are colored based on the maximum normal force. The samples values are symmetrically distributed for the gust starting position x_0 and the gust length L_g which follow symmetric distributions (see table 1) and for the gust amplitude u_e the samples are more heavily concentrated on values higher than its mean. The highest values of the maximum force occur at the highest values of the gust amplitude. A plot for the maximum tangential force shows a similar behavior.

In fig. 10 there is the least amount of variation along the gust length L_g axis. Holding L_g fixed at its mean value fig. 11 show the samples for sparse level 1,2,3 and 4. These samples are again colored based on

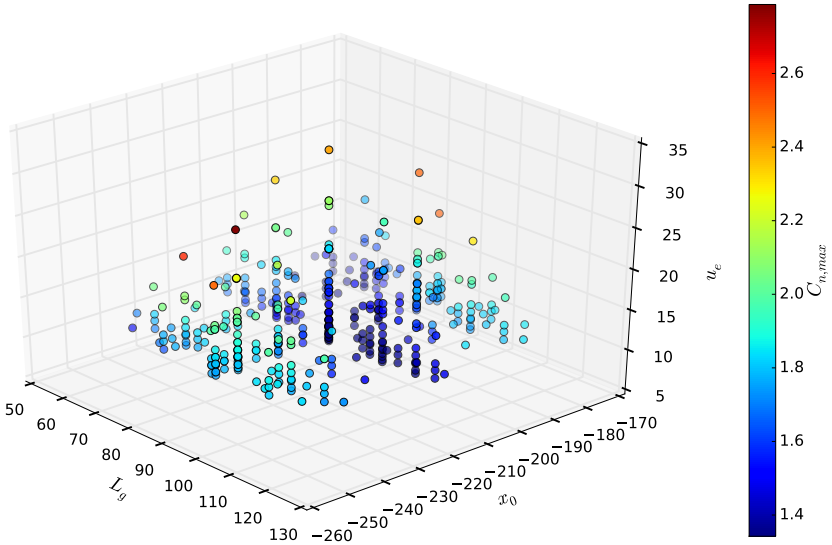


Figure 10. Sparse grid level 4 samples colored by the maximum normal force obtained by using the high-fidelity tool (SU^2).

the maximum normal force. We see that as the sparse grid level is increased there are samples with larger gust amplitude values which cause larger values for the maximum normal force. These large values can slow down the convergence of the statistics for the maximum forces.

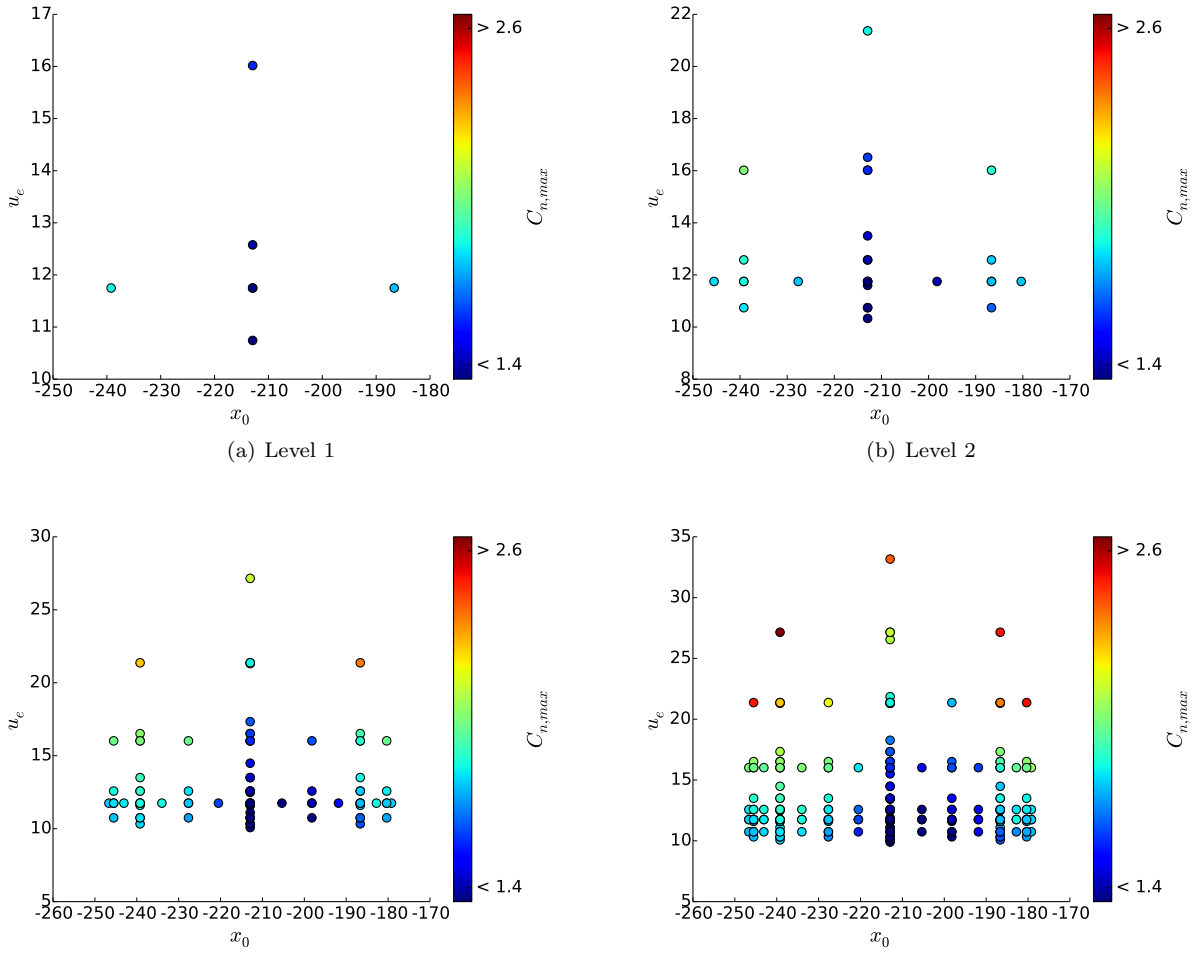
Also holding the gust amplitude fixed we can visualize how $C_{n,max}$ changes as a function of the gust starting position for the high and low-fidelity tool and also visualize what the correction function (difference between the high and low-fidelity) might look like for performing the multi-fidelity uncertainty quantification (fig. 12). For this slice of the data, the complexity of the correction function is similar to the high-fidelity and the low-fidelity does not fully follow the the high-fidelity trend, which are not favorable conditions to see large computational savings when using multi-fidelity UQ.

A convergence of the mean and the variance of the maximum forces as a function of sparse grid level for low, high and multi-fidelity is shown in figs. 13 and 14. In the multi-fidelity case the sparse grid level corresponds to the level used to approximate the correction function and the multi-fidelity uses a sparse grid level 5 for the low-fidelity. The low fidelity under predicts the value of the mean and the variance for both the maximum normal and tangential force. The multi-fidelity and high-fidelity converge towards each other which is desired. The multi-fidelity seems that it could be beneficial in providing better statistics estimates than the high-fidelity when only a small number of samples are available as the multi-fidelity values at sparse grid level 1 are closer to the values the statistics of the high and multi-fidelity are converging too for the majority of the cases.

For this problem we observe that simulating the gust at the mean values for all its parameters, the "most likely" deterministic gust is a poor estimate of the mean response for the maximum normal force figs. 13(a) and highlights the importance of properly quantifying the effect of an uncertain gust.

V. Conclusions

In this paper we have explored the influence of a uncertain extreme gust on the maximum normal and maximum tangential force on the blades of a VAWT. The extreme gust is defined by three uncertain variables, the gust starting position x_0, the gust length L_g, and the gust amplitude u_e. The maximum forces are most susceptible to the gust amplitude and the gust starting position. We studied the effect of the gust with a low-fidelity tool CACTUS (blade element/vortex lattice aerodynamic solver) and a high-fidelity tool SU^2 (CFD solver). To study the effect of a gust in the CFD solver, we developed a practical method that allows



(a) Level 1 (b) Level 2

(c) Level 3 (d) Level 4

Figure 11. Maximum force response to the uncertain variables x_0 and u_e. The gust length L_g is hold constant at its mean value.

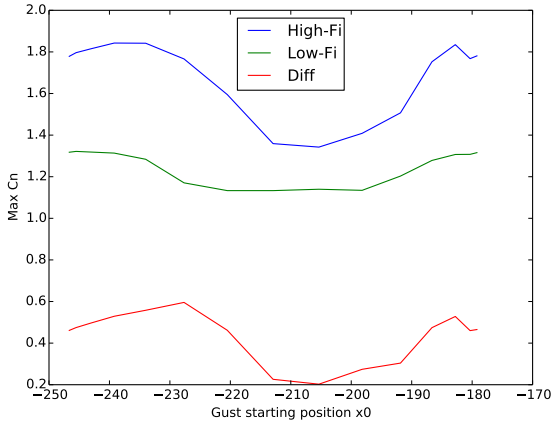
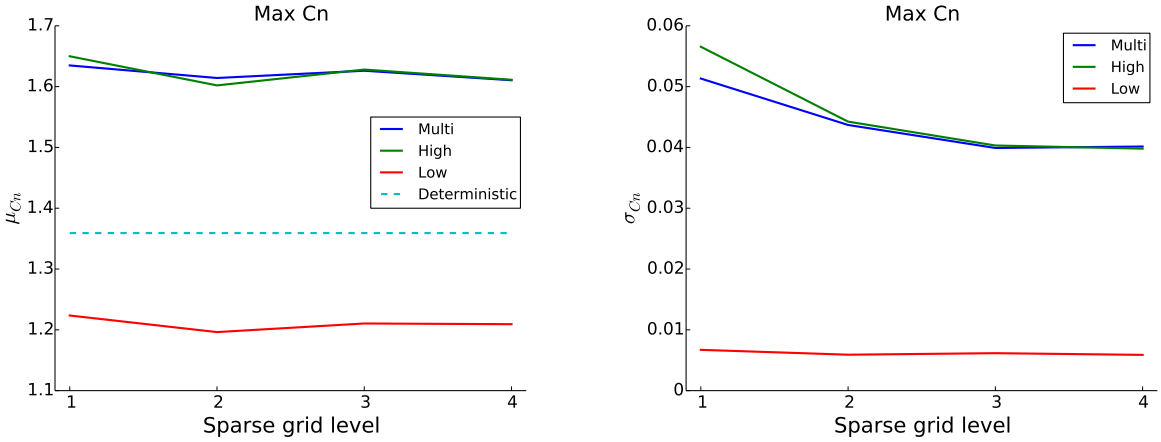
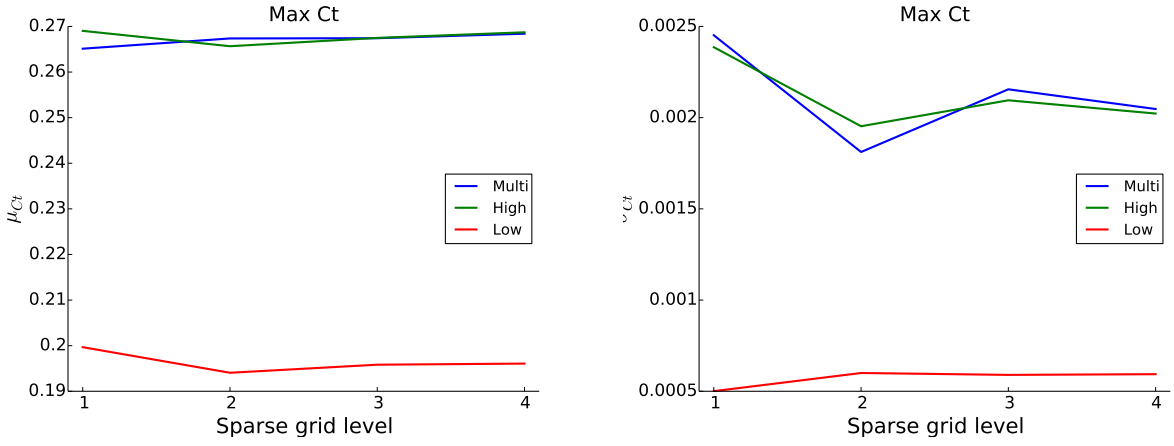


Figure 12. Maximum force response as a function of gust starting position.



(a) Mean (b) Variance

Figure 13. Convergence of maximum normal force statistics.



(a) Mean (b) Variance

Figure 14. Convergence of the maximum tangential force statistics.

us to simulate a gust that changes its magnitude in the flow direction (non-divergent free gust) by combining the field velocity method (FVM) and the geometric conservation law (GCL).

We found that simulating the gust with the low-fidelity tool underestimates the value of the mean and the variance for the maximum forces on the blade when compared to the high-fidelity tool ("true solution"). Also, simulating the gust at its deterministic "most likely" realization under predicts the maximum forces on the blade when compared to the mean values calculated by the uncertainty quantification study. A reasonable estimate of the high-fidelity mean and variance for the maximum forces is obtained in only a few (9 - sparse grid level 1) high-fidelity simulations. It is possible to improve the estimate of the maximum force statistics at a few samples by using the proposed multi-fidelity uncertainty quantification methodology. The multi-fidelity methodology uses polynomial chaos to create an approximation to the high-fidelity statistics via a correction function based on the difference between the high and low-fidelity simulations.

Acknowledgments

A. Santiago Padrón would like to acknowledge the support provided by Sandia National Laboratories through the Sandia National Laboratories/Stanford University Excellence in Engineering Research Assistantship Program award.

References

[1] "IEC 61400-1 : Wind turbines - Part 1: Design requirements," Tech. rep., International Electrotechnical Commission, 2005.

[2] Tangler, J. and Kocurek, D., "Wind Turbine Post-Stall Airfoil Performance Characteristics Guidelines for Blade-Element Momentum Methods," *43rd AIAA Aerospace Sciences Meeting and Exhibit*, American Institute of Aeronautics and Astronautics, Reston, Virigina, Jan. 2005, pp. 1–10.

[3] Islam, M., Ting, D. S.-K., and Fartaj, A., "Aerodynamic models for Darrieus-type straight-bladed vertical axis wind turbines," *Renewable and Sustainable Energy Reviews*, Vol. 12, No. 4, 2008, pp. 1087–1109.

[4] Scheurich, F., Fletcher, T., and Brown, R., "Simulating the aerodynamic performance and wake dynamics of a vertical axis wind turbine," *Wind Energy*, 2011.

[5] Scheurich, F. and Brown, R., "Modelling the aerodynamics of vertical axis wind turbines in unsteady wind conditions," *Wind Energy*, 2012, pp. 91–107.

[6] Veldkamp, D., *Chances in Wind Energy: A Probabilistic Approach to Wind Turbine Fatigue Design*, Ph.D. thesis, Tecnical University of Delft, 2006.

[7] Larsen, G. C. and Hansen, K. S., "Rational calibration of four IEC 61400-1 extreme external conditions," *Wind Energy*, Vol. 11, No. 6, Nov. 2008, pp. 685–702.

[8] Ng, L. W.-T. and Eldred, M., "Multifidelity Uncertainty Quantification Using Non-Intrusive Polynomial Chaos and Stochastic Collocation," *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA*, No. April, American Institute of Aeronautics and Astronautics, Reston, Virigina, April 2012, pp. 1–17.

[9] Korobenko, a., Hsu, M.-C., Akkerman, I., and Bazilevs, Y., "Aerodynamic Simulation of Vertical-Axis Wind Turbines," *Journal of Applied Mechanics*, Vol. 81, No. 2, Sept. 2013, pp. 021011.

[10] Li, C., Zhu, S., Xu, Y.-l., and Xiao, Y., "2.5D large eddy simulation of vertical axis wind turbine in consideration of high angle of attack flow," *Renewable Energy*, Vol. 51, March 2013, pp. 317–330.

[11] Takahashi, S., Hamada, J., and Takashi, Y., "Numerical and experimental studies of airfoils suitable for Vertical Axis Wind Turbines and an application of wind-energy collecting structure for higher performance," *Journal of Wind Engineering*, 2006, pp. 327–330.

[12] Palacios, F., Alonso, J., Duraisamy, K., Colonno, M., Hicken, J., Aranake, A., Campos, A., Copeland, S., Economon, T., Lonkar, A., Lukaczyk, T., and Taylor, T., "Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design," *51st AIAA Aerospace Sciences Meeting*, No. January, American Institute of Aeronautics and Astronautics, Reston, Virigina, Jan. 2013, pp. 1–60.

[13] Palacios, F., Duraisamy, K., Aranake, A., Copeland, S. R., Economon, T. D., Lonkar, A. K., Lukaczyk, T. W., Naik, K. R., Padron, S., and Alonso, J. J., "Stanford University Unstructured (SU2): Analysis and Design Technology for Turbulent Flows," *52nd Aerospace Sciences Meeting*, No. January, American Institute of Aeronautics and Astronautics, Reston, Virginia, Jan. 2014, pp. 1–33.

[14] Jameson, A., Schmidt, W., and Turkel, E., "Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes," *AIAA paper*, Vol. M, 1981, pp. 1–19.

[15] Jameson, A., "Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings," AIAA, 1991.

[16] Murray, J. and Barone, M., "The Development of CACTUS, a Wind and Marine Turbine Performance Simulation Code," *49th AIAA Aerospace Sciences Meeting*, No. January, American Institute of Aeronautics and Astronautics, Reston, Virigina, Jan. 2011, pp. 1–21.

[17] Raveh, D. E., "CFD-Based Models of Aerodynamic Gust Response," *Journal of Aircraft*, Vol. 44, No. 3, May 2007, pp. 888–897.

[18]Parameswaran, V. and Baeder, J. D., "Indicial Aerodynamics in Compressible Flow-Direct Computational Fluid Dynamic Calculations," *Journal of Aircraft*, Vol. 34, No. 1, Jan. 1997, pp. 131–133.

[19]Singh, R. and Baeder, J. D., "Direct Calculation of Three-Dimensional Indicial Lift Response Using Computational Fluid Dynamics," *Journal of Aircraft*, Vol. 34, No. 4, July 1997, pp. 465–471.

[20]Zaide, A. and Raveh, D., "Numerical Simulation and Reduced-Order Modeling of Airfoil Gust Response," *17th AIAA Computational Fluid Dynamics Conference*, No. June, American Institute of Aeronautics and Astronautics, Reston, Virigina, June 2005, pp. 1–16.

[21]Bartels, R., *Development, Verification and Use of Gust Modeling in the NASA Computational Fluid Dynamics Code FUN3D*, No. October, 2012.

[22]Wales, C., Jones, D., and Gaitonde, A., "Reduced order modelling for aeroelastic aerofoil response to a gust," *51st AIAA Aerospace Sciences Meeting*, No. January, American Institute of Aeronautics and Astronautics, Grapevine, Texas, 2013, pp. 1–16.

[23]BATINA, J. T., "Unsteady Euler airfoil solutions using unstructured dynamic meshes," *AIAA Journal*, Vol. 28, No. 8, Aug. 1990, pp. 1381–1388.

[24]Lesoinne, M. and Farhat, C., "Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations," *Computer methods in applied mechanics and . . .*, Vol. 7825, No. 96, 1996.

[25]Farhat, C., Geuzaine, P., and Grandmont, C., "The Discrete Geometric Conservation Law and the Nonlinear Stability of ALE Schemes for the Solution of Flow Problems on Moving Grids," *Journal of Computational Physics*, Vol. 174, No. 2, Dec. 2001, pp. 669–694.

[26]Xiu, D. and Karniadakis, G., "The Wiener–Askey polynomial chaos for stochastic differential equations," *SIAM Journal on Scientific Computing*, Vol. 02912, 2002, pp. 1–26.

[27]Witteveen, J. A. and Bijl, H., "Modeling Arbitrary Uncertainties Using Gram-Schmidt Polynomial Chaos," *44th AIAA Aerospace Sciences Meeting and Exhibit*, No. January, American Institute of Aeronautics and Astronautics, Reston, Virigina, Jan. 2006.

[28]Golub, G. and Welsch, J., "Calculation Of Gauss Quadrature Rules," *Mathematics of Computation*, 1969.

[29]Gerstner, T. and Griebel, M., "Numerical integration using sparse grids," *Numerical algorithms*, Vol. 18, 1998, pp. 209–232.

[30]Eldred, M. S., "Design Under Uncertainty Employing Stochastic Expansion Methods," *International Journal for Uncertainty Quantification*, Vol. 1, No. 2, 2011, pp. 119–146.