POD-ACCELERATED CFD ANALYSIS OF WIND LOADS ON PV SYSTEMS

Victor Huayamave[1,3], Andres Ceballos[1,3], Eduardo Divo[2], Alain Kassab[3],
Stephen Barkaszi[4], Hubert Seigneur[4], Carolina Barriento[1]

[1]Central Technological Corporation
932 Larson Rd, Altamonte Springs, FL 32714, USA

[2]Department of Mechanical Engineering
Embry-Riddle Aeronautical University
600 South Clyde Morris Blvd, Daytona Beach, FL 32114, USA
e-mail: divoe@erau.edu

[3]Department of Mechanical and Aerospace Engineering
University of Central Florida
4000 Central Florida Blvd, Orlando, FL 32816, USA

[4]Florida Solar Energy Center
University of Central Florida
1679 Clearlake Rd, Cocoa Beach, FL 32922, USA

ABSTRACT

A real-time response framework based on the Proper Orthogonal Decomposition (POD) method is proposed to provide a solution that would not only take advantage of the great detail and accuracy of a grid-converged 3D computational fluid dynamics (CFD) analysis but also calculate, in real-time, flow features and loads that result from wind-induced drag and lift forces on Photo-Voltaic (PV) systems. The key is to generate beforehand and off-line an extensive set of solutions, i.e. pressure and shear stress distributions over the PV system surface, using CFD within a predefined design space (module sizes, wind speeds, topographies, roof dimensions, pitch, etc.). These solutions are then organized to form the basis snapshots of a POD decomposition matrix. An interpolation network using radial-basis functions (RBF) will be employed to predict the solution from the POD decomposition given a set of values of the design variables. The entire POD matrix and RBF interpolation network are stored in a database that can be accessed remotely by the wind-load calculator tool and therefore predict the solution, flow features and loads, in real time. The trained POD-RBF acts as a multifaceted interpolation that preserves the physics of the problem and has been tested and validated by performing the fast algebraic interpolation to obtain the pressure distribution on the PV system surface and comparing them to actual grid-converged fully-turbulent 3D CFD solutions at the specified values of the design variables (wind speed and angle).

NOMENCLATURE

p	[psf]	Pressure field
V	[mph]	Velocity field
μ	[Pa-s]	Dynamic Viscosity
ρ	[kg/m^3]	Density
U	[-]	POD Snapshot matrix
C	[-]	POD Covariance matrix
Φ	[-]	POD Basis
V	[-]	Eigenvectors
Λ	[-]	Eigenvalues
A	[-]	POD Amplitudes
P	[-]	Design parameter space
$f(\mathbf{p})$	[-]	Radial-basis function (RBF)
c	[-]	RBF smoothing parameter
F	[-]	Interpolation matrix
B	[-]	Interpolation coefficients

INTRODUCTION

Wind loading calculation for structures are currently performed according to the ASCE (American Society of Civil Engineer) 7 standard. The values in this standard were calculated from simplified models that do not necessarily take into account relevant characteristics such as those from full 3D effects, end effects, turbulence generation and dissipation, as well as minor effects derived from shear forces on installation brackets and other accessories. This standard does not have provisions to handle the majority of rooftop PV systems, and attempts to apply this standard may lead to significant design

errors as wind loads are incorrectly estimated. Therefore, an accurate calculator would be of paramount importance for the preliminary assessments of the uplift and down force loads on a PV mounting system, identifying viable solutions from available mounting systems, and therefore helping reduce the cost of the mounting system installation by streamlining the process. The challenge is that although a full-fledged 3D CFD analysis would properly and accurately capture the complete physical effects of air flow over PV systems, it would be unsuitable for this tool, which is intended to be a real-time calculator. This is because in order to arrive at a solution that can be deemed accurate and grid-independent, CFD routinely requires enormous computation times even in powerful and massively parallel computer platforms.

Although CFD is a well-established and proven accurate analysis tool, a major drawback of CFD analysis is that it requires meticulously structured and generally static meshes in order to arrive at accurate solutions; meshes which demand significant human interaction time to construct. For this reason much of the effort expended in current CFD analysis is directed toward the development of suitable CFD grids. The lack of automation in this mesh generation process also poses significant limitation on the usefulness of CFD analysis for the system design. Development of a truly automated, robust CFD algorithm for viscous flow applications is a vital step toward parametric design optimization.

The solution approach is based on the POD method to build a fast response framework that laps into the power and precision of CFD analysis without requiring its extensive setup times and massive computational requirements, while preserving the physics and thus the accuracy of the solution. The concept of POD began over a century ago as a statistical tool developed by Pearson [1]. Since that time, this method has been redeveloped under various names and in vastly different applications. Depending on how the input data is utilized POD is also similarly known as Karthunen-Loeve decomposition (KLD), principal component analysis (PCA) or singular value decomposition (SVD) [2][3]. Furthermore, this technique has been implemented in various applications from signal processing and control theory, human face recognition, data compression, fluid mechanics, parameter estimation and many others. The POD can be used to produce a low-order, but high quality, approximation of the solution field. More specifically POD is often capable of capturing dominant components (called principal components) of the data with typically only a few modes. This is due to the ability of POD to offer the best basis for least-squares approximation defining a set of vectors using a rotated coordinate frame, where the angles of rotation are denoted as the POD basis [4][5]. The integration of POD into design and optimization problems arose due to the demanding task of repeatedly solving forward problems while varying certain parameters in the process of seeking the solution of an ill-posed inverse problem. The POD method capitalizes on the correlation between the known direct problem and the sought-after solution [6][7]. Of course, the primary reason POD is a favourable in solving design and optimization problems, is that it provides many desired features

such as model reduction, error filtration and regularization. An application involving design and optimization coupling the POD and CFD analysis was recently reported in the literature [8] showing significant solution acceleration and adaptability. More recently the method was applied to the inverse problem of parameter estimation in mechanics and reported in [9].

BACKGROUND

To perform CFD analysis over the parameterized set of PV systems, the incompressible viscous flow equations that govern the behaviour of air flow must be solved. These are the incompressible Navier-Stokes equations, which, in the Eulerian frame of reference, have the following form:

$$\nabla \cdot \vec{V} = 0$$
$$\rho \frac{\partial \vec{V}}{\partial t} + \rho \left(\vec{V} \cdot \nabla \right) \vec{V} = -\nabla p + \mu \nabla^2 \vec{V} \tag{1}$$

Where ρ is the density and is the μ dynamic viscosity. The field variable are the velocity field, $\vec{V}$, and the pressure, p. All field variables are functions of space and time, in a fixed domain Ω surrounded by a closed boundary Γ. The explicit space-time dependency of each dependent variable has been omitted for simplicity of notation.

To build the POD and interpolation network the first step in the implementation of the POD is the creation of the snapshot, which is the collection of N sample values of **u** (field under consideration). In this case, **u** stores the discrete values of wall shear stress and pressure. Next, a collection of M snapshots denoted as $\mathbf{u^j}$ (for j = 1, 2 ... M) are generated by altering the parameter(s) upon which the field depends on. In the current scope, these refer to the module configuration, the pitch, and the wind velocities. Each $\mathbf{u^j}$ is the stored inside rectangular N x M matrix **U** denoted as the snapshot matrix. The goal of POD is to establish a set of orthonormal vectors $\mathbf{\Phi^j}$ (for j = 1, 2 ... M) resembling the snapshot matrix **U** in an optimal way. The matrix $\mathbf{\Phi}$ is commonly referred to as the POD basis and is given by:

$$\mathbf{\Phi} = \mathbf{U} \cdot \mathbf{V} \tag{2}$$

Where **V** represents the eigenvectors of the covariance matrix **C** and can be easily be derived using the nontrivial solution of the general eigenvalue problem denoted as:

$$\mathbf{C} \cdot \mathbf{V} = \Lambda \cdot \mathbf{V} \tag{3}$$

Λ represents a diagonal matrix that stores the eigenvalues λ of the covariance matrix **C**, which is defined as $\mathbf{C} = \mathbf{U^T} \cdot \mathbf{U}$. It may also serve to note that **C** is symmetric and positive definite and λ is always real and positive. Typically λ is sorted in a descending order and can often be attributed to the energy of the POD mode (base vector). This energy decreases rapidly with the increasing mode number. Since higher modes hold little energy (or data) of the system they can be discarded

without influencing the accuracy of representation. This is known as the truncation of the POD basis and is accomplished by deciding which fraction of the energy of the system can be neglected in later calculations. The resulting POD basis $\mathbf{\Phi}$, referred to as the truncated POD basis consist of K < M vectors and is given by:

$$\hat{\mathbf{\Phi}} = \mathbf{U} \cdot \hat{\mathbf{V}} \qquad (4)$$

This also corresponds to the truncation of the eigenvector matrix, denoted as $\hat{\mathbf{V}}$, which stores the first K^{th} eigenvectors of $\mathbf{C}$. The truncated POD basis is also known to be orthogonal $\mathbf{\Phi}^T \cdot \mathbf{\Phi} = \mathbf{I}$ and presents optimal approximation properties. Once $\hat{\mathbf{\Phi}}$ is known, the snapshot matrix $\mathbf{U}$ can be regenerated and approximated as $\mathbf{U} = \hat{\mathbf{\Phi}} \cdot \mathbf{A}$, where $\mathbf{A}$ stands for the amplitudes associated with u^j. Now referring to the orthogonality of $\hat{\mathbf{\Phi}}$, the amplitudes can be determined from $\mathbf{A} = \hat{\mathbf{\Phi}}^T \cdot \mathbf{U}$.

At this time, data may begin to be extrapolated for information on the current problem. To do this, consider a vector $\mathbf{p}$, which stores the parameters on which the solution depends. Next, the amplitudes $\mathbf{A}$ are defined as a nonlinear function of the parameter vector $\mathbf{p}$. The unknown constant coefficients of the current combination are gathered in a matrix $\mathbf{B}$, as $\mathbf{A} = \mathbf{B} \cdot \mathbf{F}$, where $\mathbf{F}$ is defined as the matrix of interpolation functions, where the set of interpolation functions $f_i(\mathbf{p})$ can be chosen arbitrarily. However, some choices of interpolation functions may lead to an ill-conditioned system of equations for the coefficient matrix $\mathbf{B}$. For instance, Radial-basis functions (RBF) can be used as the interpolating function of choice due to their nice approximation and smoothing properties. One of such RBFs is the Hardy inverse Multiquadric [10][11]:

$$f_i(\mathbf{p}) = f_i\left(\left|\mathbf{p} - \mathbf{p}^i\right|\right) = \frac{1}{\sqrt{\left|\mathbf{p} - \mathbf{p}^i\right|^2 + c^2}} \qquad (5)$$

Where c is defined as the RBF smoothing factor and $\mathbf{p}^i$ corresponds to the same parameter $\mathbf{p}$ used to generate $\mathbf{u}^i$ (for i = 1, 2 … M). The matrix of coefficients $\mathbf{B}$ can be evaluated by simple inversion as $\mathbf{B} = \mathbf{A} \cdot \mathbf{F}^{-1}$, where $\mathbf{F}$ is the matrix of interpolation functions defined as set of M identical vectors $\mathbf{f}(\mathbf{p})$ defined as $\{\mathbf{f}\}_j = f_j\left(\left|\mathbf{p} - \mathbf{p}^j\right|\right)$. At this point it should be stressed that the matrix of amplitudes $\mathbf{A}$ and the matrix of coefficients $\mathbf{B}$ are known using the above relations. Now the following equation is arrived at:

$$\mathbf{\Phi}^T \cdot \mathbf{U} = \mathbf{B} \cdot \mathbf{F} \qquad (6)$$

Using the orthogonality of $\hat{\mathbf{\Phi}}$, it can easily be seen that the snapshot matrix $\mathbf{U}$ can be approximated as $\mathbf{U}(\mathbf{p}) \approx \mathbf{\Phi} \cdot \mathbf{B} \cdot \mathbf{F}(\mathbf{p})$, such that after the coefficient matrix $\mathbf{B}$ is evaluated, a low dimensional model can be set in vector form as:

$$\mathbf{u}(\mathbf{p}) \approx \hat{\mathbf{\Phi}} \cdot \mathbf{B} \cdot \mathbf{f}(\mathbf{p}) \qquad (7)$$

This model will now be referred to as the *trained* POD-RBF network and is completely capable of reproducing the unknown field that corresponds to any arbitrary set of parameters $\mathbf{p}$. This can be thought of as a numerical eingenfunction expansion of the solution reminiscent of the variation of parameters (or integral transform) method for the analytical solution of partial differential equations.

PARAMETERIZATION OF THE PV SYSTEM CONFIGURATIONS AND SOLID MODELING

To create the fluid domain for CFD analysis, several geometrical configurations were created to represent the roof-mounted PV systems. Sets of representative values of the design variables or parameters were selected so as to cover a wide design space. That is, selected standard PV module configuration arranged in a rectangular fashion as well as a discrete set of pitch angles between 18 degrees, 22.5 degrees, and 26.5 degrees. The angles were selected because they are a subset of typical roof slopes used in ASCE 7 design tables. The solid models were built using SolidWorks® (Dassault Systemes, Concord, MA) and the fluid domain was extracted so that it extends far away from the region of influence of the system

Commercial PV Modules

PV modules currently used for residential installations have power output of 200 to 260 Watts. This class of modules has dimensions of 60" to 66" by 36" to 39" and a frame thickness of 1.25" to 2". Given these ranges, nominal module dimensions were selected to be 60" x 39" x 1.5".

Arrangement of Solar Panels

The number of modules selected for the PV array was arbitrarily selected but based on common PV array configurations and the layout was partially constrained by the building dimensions. A total of 27 modules were arranged in 3 rows and 9 columns. A system with this number of modules would have a nominal output rating of 5 to 7 kW. This is also in the range of typical residential installations.

Standard house roof angles

The base case model is configured as a single-story building with a rectangular base and a gable roof. Three different roof slopes (18 degrees, 22.5 degrees, 26.5 degrees) were selected for the initial evaluations. These represented roof angles used in the ASCE 7 wind load tables and are common roof slopes found in high wind regions. The final configuration was then arbitrarily modeled using the standard 26.5 degree (6:12 gable roof).

CFD ANALYSIS OVER THE PARAMETRIZED SET OF PV SYSTEMS

The fluid domains of the solid models generated were then discretized using a combination of tetrahedral and hexahedral

cells and tested for grid-convergence using Star-CCM+. A realizable κ-ω turbulent model was implemented and commonly used values of inlet turbulent intensity were imposed. The wind speed was introduced as a third design variable or parameter. The grid-converged CFD solution over each of the parameterized PV system was then post-processed to integrate the wall shear stress and pressure over the entire surface of the system to yield the sought after values of wind-load (uplift or downforce). Due to the nonlinear nature of the fluid flow phenomenon, it is expected that the wind-loads will exhibit a highly nonlinear relationship with respect to the design parameters and possibly a non-monotonic one that can lead to multiple minima and/or maxima.

Computational Domain

The parameterized model was used to create the computational domain. In order to obtain fully developed flow and capture flow features downstream and in the vicinity of the house/PV panel assembly, a computational domain was built using a bounding rectangular box of the dimensions depicted in Figure 1. The dimensions correspond to the following locations in reference to a coordinate system placed at the geometric center of the base of the house. The hydraulic diameter (HD) is the chord length of the house (360"), the inlet placed 5 HD upstream (-X) of the origin, the outlet placed 10 HD downstream (+X) of the origin, the sides placed 5 HD at either side (-Z, Z+) of the origin, and the top placed 5HD above (Y+) the origin.

Meshed Computational Domain

After the computational domain was completed, the mesh was generated using predominantly hexahedral cells with trimmed cells near the boundaries. The use of thin, prismatic cells near the wall boundaries provides appropriate boundary layer resolution required in order to make use of the κ-ω SST turbulence model. Volumetric mesh controls were used to refine the areas near the house and panels. The final mesh was arrived at through a grid-convergence study with a number of about 10 million cells.
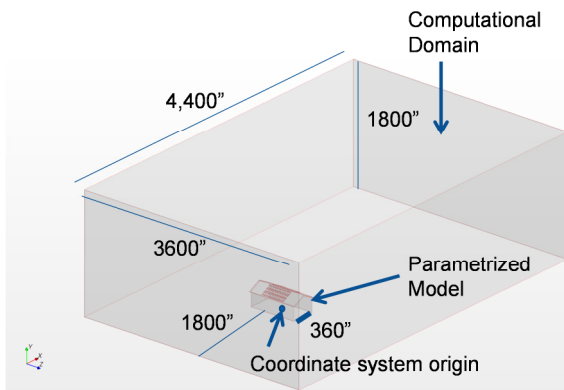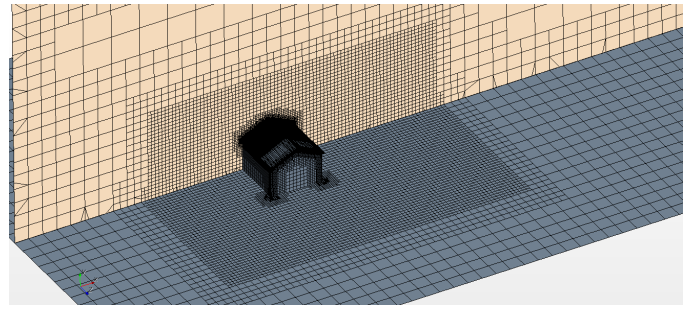

Figure 1 Computational domain dimensions.

Figure 2 Meshed computational domain.

Physics Models

For the analysis, an incompressible flow model with κ-ω SST turbulence with two-layer, all y+ wall treatment scheme was used. One initial case was run with the κ-ε model to initialize subsequent runs with the κ-ω model. The κ-ω SST model offers improved performance for boundary layers under adverse pressure relative to the κ-ε model. The SST model has seen fairly wide application in the aerospace industry, where viscous flows are typically resolved and turbulence models are applied throughout the boundary layer. Potentially more accurate models such as Large Eddy Simulation (LES) and Detached Eddy Simulation (DES) are not considered for the time being since they require excessively higher mesh resolution and computational expense.

The analysis was initially run in steady state, after which the inherently unstable nature of the flow prompted the use of an unsteady (transient) solver. A first-order implicit unsteady solver was used with a second-order up-winding scheme for convective derivatives and flux limiters in all cases.

Boundary Conditions

Once the control volume was meshed, the boundary conditions were applied to the model. An operating pressure of 1 atm, an uniform inlet velocity profile (provides maximum stresses on roof), a density of 1.18415 kg/m^3, a viscosity of 1.85508e-05 Pa-s, a 5% turbulent intensity, a non-reflective pressure outlet, and zero gauge total pressure were applied. Also a smooth wall side condition to the ground and symmetry to the top of bounding box were applied.

Validation of POD Interpolation Network

The first step in the implementation of the POD is the creation of the *snapshot* which is the collection of N sampled values of *u* - the field under consideration. In this case, *u* stores the discrete values of stress (normal and shear). The panel configuration consists of 9x3 PV modules arranged as shown in Figure 3.
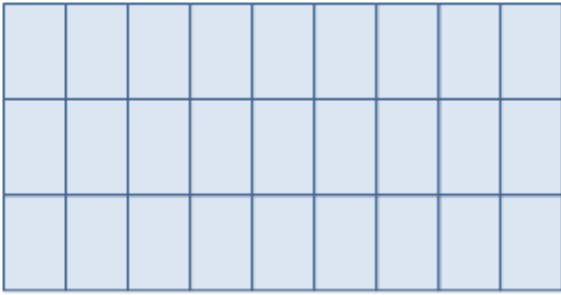
Figure 3 Panel configuration of 9x3 standard PV modules.

Each standard PV module is 39"x60", so a point distribution for POD sampling was setup with 13x20 points per PV module separated 3" with a margin of 1.5". This point configuration and spacing is shown in Figure 4. Therefore the number of sampled values of wall shear stress and pressure on the panel is $N=(9x3)x(13x20)=7,020$. This N value provides sufficient resolution for the wall shear stress and pressure distribution on the panel as well as for its integration to render the lift force.

Next, a collection of M snapshots denoted as $\mathbf{u}^j$ (*for j = 1, 2 … M*) are generated by altering the parameters upon which the field depends on. In the current scope, these refer to the wind speed (80mph to 200mph, in 20mph increments) and wind angle (360° around, in 30° increments) for a total number of snapshots generated of $M=91$. Each $\mathbf{u}^j$ is then stored inside a rectangular $N \ x \ M$ matrix $\mathbf{U}$ denoted as the snapshot matrix. Note that the dimensions of the snapshot matrix $\mathbf{U}$ are 7,020x91.
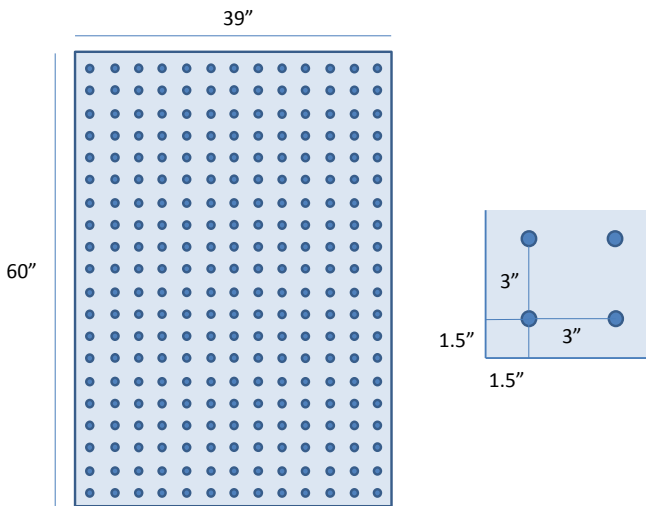

Figure 4 Point distribution and spacing on PV module for POD sampling.

RESULTS
CFD Results
A total of ninety-one (91) configurations were defined by altering two input parameters: wind speed (80mph to 200mph, in 20mph increments) and wind angle (360° around, in 30° increments). The number of CFD runs was reduced to only twenty-eight (28) by taking advantage of symmetry and by placing PV panels on either side of the gable roof in order to render two or four solutions in one CFD run.

Figure 5 shows plots of representative streamlines colored by velocity magnitude and pressure contours on the panel surface, roof, and ground. The results shown in Figure 5 correspond to a wind speed of 100mph and wind angles of 0˚, 30˚ (or 330˚), and 60˚ (or 300˚). The plots were rendered at a specific time value of the time-accurate CFD analysis when the total force generated on the panel (monitored during the CFD run) was at a maximum value. Notice the clear distinction of the pressure gradient on the PV system surface as well as its nonlinear behavior.
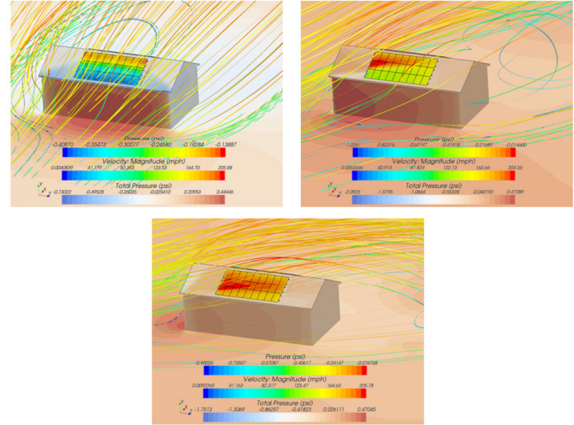

Figure 5 Streamlines colored by velocity and pressure contour plot of panel assembly, roof, and ground at 100mph wind speed. Configurations are: 0˚ (Top Left), 30˚ (Top Right), 60˚ (Bottom).

The analysis configuration schedule is shown in Table 1 with the resulting values of average pressure in *psf*. Again, these values of average pressure on the PV system were rendered at a specific time value of the time-accurate CFD analysis when the total force generated on the panel (monitored during the CFD run) was at a maximum value.

POD Results
After the CFD computations were performed for all cases and the 7,020x91 POD snapshot matrix $\mathbf{U}$ was formed, the decomposition was performed and tested. The 91x91 covariance matrix $\mathbf{C}$ was formed as $\mathbf{C} = \mathbf{U^T \cdot U}$ followed by a standard eigenvalue decomposition which produced the results shown in Figure 6. Note that the 91 eigenvalues are displayed in logarithmic scale showing a maximum value of about 10^9 and a minimum value of about 10^{-5}. More importantly, the eigenvalues decrease very rapidly from the largest value to less than 10^3 after the first 12 eigenvalues, indicating that most of the system information (energy) is contained and can be extracted from the first few eigenvalues using a truncated POD basis.

Table 1 CFD analysis configuration schedule

Pressure (psf) Speed\Angle	0˚	30˚ (330˚)	60˚ (300˚)	90˚ (270˚)
80 mph	-10.39	-8.89	-5.44	-7.57
100 mph	-16.25	-13.89	-8.43	-11.83
120 mph	-23.42	-20.01	-12.15	-17.03
140 mph	-31.90	-27.24	-16.74	-23.17
160 mph	-41.69	-35.59	-21.83	-30.26
180 mph	-52.79	-45.05	-27.60	-38.29
200 mph	-65.20	-55.63	-34.50	-47.26
Pressure (psf) Speed\Angle	120˚ (240˚)	150˚ (210˚)	180˚	
80 mph	-8.36	-8.35	-6.44	
100 mph	-13.07	-13.05	-10.05	
120 mph	-18.90	-18.79	-14.47	
140 mph	-25.91	-25.58	-19.69	
160 mph	-33.87	-33.42	-25.72	
180 mph	-43.46	-42.31	-32.55	
200 mph	-53.76	-52.25	-40.19	
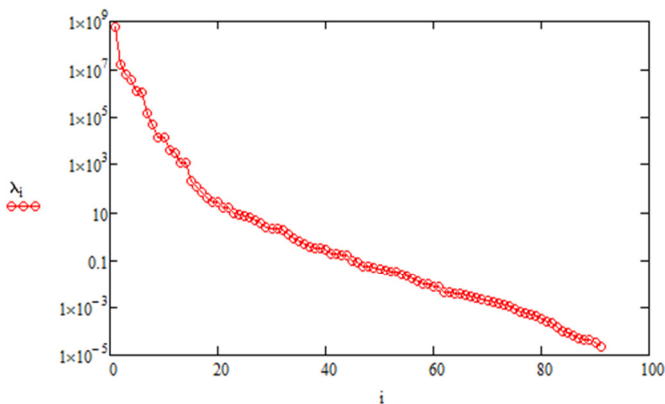
Figure 6 Eigenvalues of the POD covariance matrix sorted in descending order.

A truncated POD basis $\hat{\Phi}$ was built by using only the first 12 eigenvalues and eigenvectors $\hat{V}$ as $\hat{\Phi} = U \cdot \hat{V}$. Then the POD snapshot matrix was reconstructed as $U_w = \hat{\Phi} \cdot A$ where the amplitudes are: $A = \hat{\Phi}^T \cdot U$. The pressure field is then obtained from each row of the reconstructed POD snapshot matrix U_w. To illustrate the accuracy of the POD truncation a test is performed comparing the CFD-generated pressure distribution and the POD-generated pressure distribution truncated after 12 eigenvalues for a wind speed of 140 mph and a wind angle of 30°. This comparison is shown in Figure 7(a) and 7(b) respectively revealing an almost perfect qualitative accuracy. The relative RMS error between these two solutions was found to be just 0.025%, again revealing the high accuracy of the truncated POD approximation.
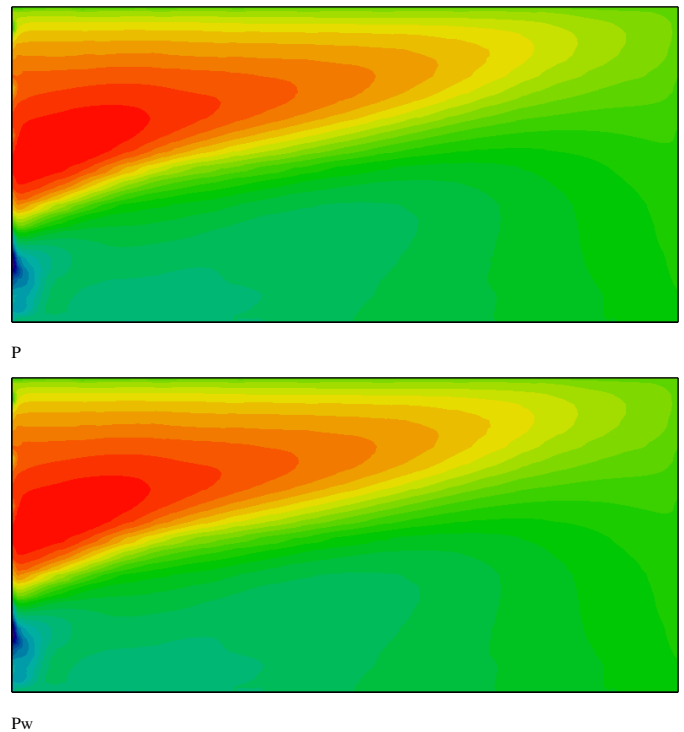

P

Pw

Figure 7 Pressure contours on PV panel at 140 mph and 30°. (a) CFD-generated and (b) Truncated POD-generated.

POD-RBF Interpolation Results

The trained POD-RBF interpolation network shown in Eqns. (5)-(7) was implemented and tested. The design parameters **p** where in this case chosen to be the wind speed $p_1=V$ and the wind angle $p_2=\beta$ in this particular order. In addition, the RBF smoothing factor c in Eqn. (5) was chosen so as to produce a well-conditioned interpolation capable of generating smooth evaluations outside the collocation points $p_1=\{80,100,\ldots,200\}$, $p_2=\{0,30,\ldots,360\}$. This is accomplished by selecting a value of c that produces an interpolation matrix **F** with a high condition number but within the range of the precision used for the floating-point representation of the variables (double-precision in this case). For this case, the smoothing factor was selected to be:

1775

$$c = \frac{1}{4}\sqrt{(200-80)^2 + (360)^2} \qquad (8)$$

This value of the smoothing factor resulted in an interpolation matrix $\mathbf{F}$ exhibiting a condition number based on the L-infinity norm of $C = 6 \cdot 10^7$, which is within the range of double-precision floating point representation.

Once the interpolation matrix $\mathbf{F}$ was formed, the interpolation coefficients $\mathbf{B}$ were obtained using the POD amplitude matrix $\mathbf{A}$ as: $\mathbf{B} = \mathbf{A} \cdot \mathbf{F}^{-1}$. These interpolation coefficients were then employed with the RBF interpolation formula in Eqn. (7) to evaluate the pressure distribution on the 7,020 points distributed on the panel at any arbitrary value of the design parameters p_1: wind speed and p_2: wind angle.

The POD-RBF interpolation network was tested using two CFD solutions that were not originally used as part of the POD snapshots. First, Figure 8(a) and 8(b) show the CFD-generated ($P_{average}$=-13.15psf) and POD-RBF-rendered ($P_{average}$=-13.12psf) pressure distribution at a wind speed of 90mph and an angle of 0°. The qualitative comparison of the two solutions shows virtually no error while a quantitative comparison through a relative RMS error reveals a difference of 0.34%, demonstrating excellent agreement between the POD-RBF-rendered pressure distribution and the CFD-generated one, with the difference that the POD-RBF network produces instant results in any platform while the CFD solution requires several hours (or days) of computation to yield a grid-converged results in a high-end cluster.
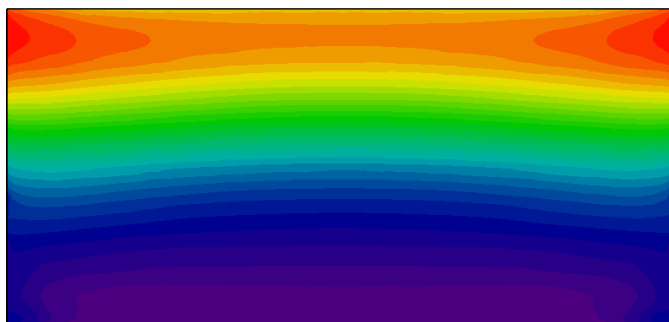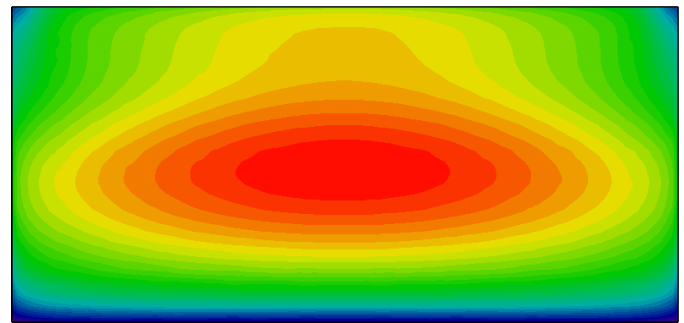
P_{CFD}
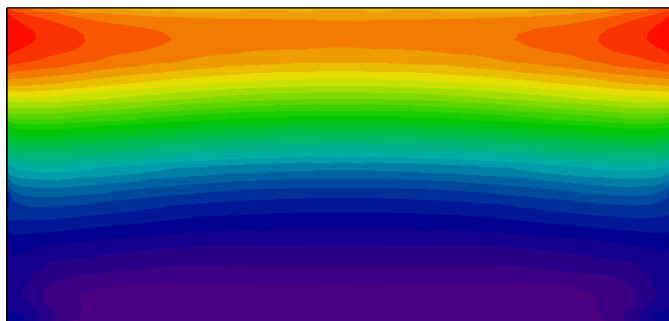
P_{POD}

Figure 8 Pressure contours on PV panel at 90 mph and 0°. (a) CFD-generated and (b) POD-RBF-generated.

The second test was carried out at a wind speed of 90mph and an angle of 180°. Figure 9(a) and 9(b) show the CFD-generated ($P_{average}$=-8.14psf) and POD-RBF-rendered ($P_{average}$=-8.14psf) pressure distribution at a wind speed of 90 mph and an angle of 0°. Again, the qualitative comparison of the two solutions shows virtually no error while a quantitative comparison through a relative RMS error reveals a difference of 0.98%, demonstrating very good agreement between the POD-RBF-rendered pressure distribution and the CFD-generated one.

These two comparison cases provide the validation and confidence necessary to implement the POD-RBF interpolation network to predict wind load distributions over PV panels at arbitrary wind velocities and angles dictated by installation requirements and codes.

P_{CFD}
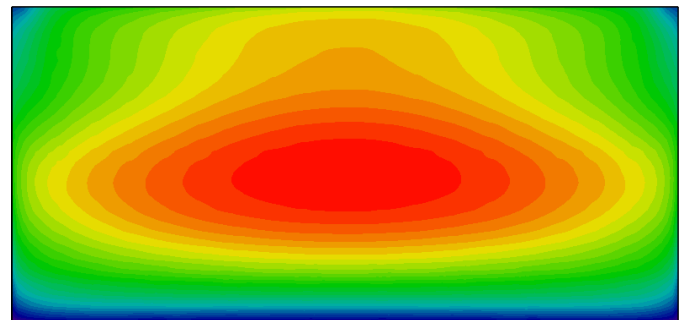
P_{POD}

Figure 9 Pressure contours on PV panel at 90 mph and 180°. (a) CFD-generated and (b) POD-RBF-generated.

Implementation in a Web-based GUI

A Graphical User Interface (GUI) for the Wind-Load Calculator has been developed to take advantage of the trained POD-RBF interpolation network in order to provide the pressure distribution on the panel at arbitrary values of wind speed and wind angle. In addition, it provides the average pressure as well as the resulting uplift force on the PV panel. This GUI was developed in *JavaScript* so that it can be rendered in HTML5 format from any Web browser on a desktop computer, laptop, tablet, or even smartphone. The layout is automatically refined to appropriately fit in desktop or mobile displays with legends as well as options for contour coloring. A rendering of the Wind-Load Calculator Interface can be seen in Figure 10.
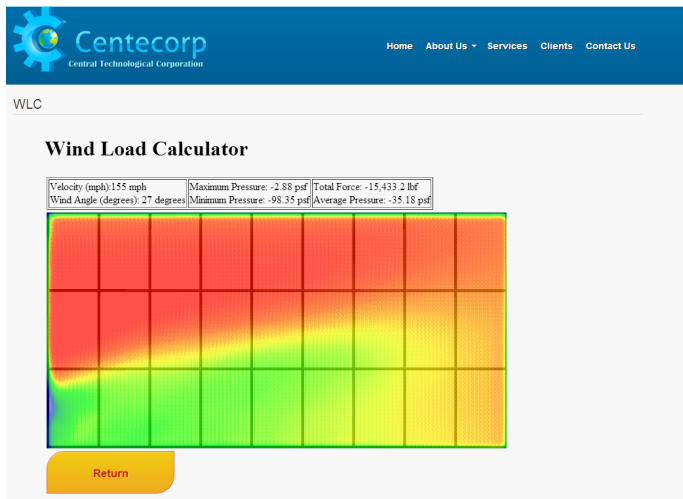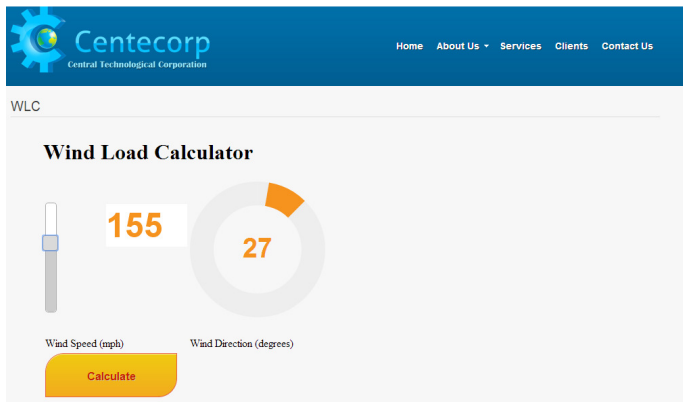
Figure 10 Rendering of the Wind-Load Calculator Interface.

CONCLUSION

The trained POD-RBF interpolation network has been tested and validated by performing the fast algebraic interpolation to obtain the pressure distribution on the PV system surface and comparing them to actual grid-converged fully-turbulent 3D CFD solutions at the specified values of the design variables (wind speed and angle). The solar power industry, engineering design firms, as well as society as a whole, could realize significant savings with the availability of a real-time in-situ wind-load calculator that can prove essential for plug-and-play installation of PV systems. The success of this project will have a direct impact on the installation process of PV systems by allowing readily-available off-the-shelf components and modules to be assembled and mounted without the need for extensive engineering analysis behind the scene. Additionally, this technology allows for automated parametric design optimization in order to arrive at the best fit for a set of given operating conditions. All these tasks are currently prohibitive due to the massive computational resources, effort, and time required to address large-scale CFD analysis problems, all made possible by a simple but robust technology that can yield massive savings for the solar power industry.

REFERENCES

[1] K. Pearson, *On lines planes of closes fit to system of points in space*, The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science, 2, (1901), 559–572.

[2] K. Karhunen, *Ueber lineare Methoden der Wahrscheinigkeitsrechnung*, Ann Acad Sci Fennicae A1 Math., Phys.,37, (1946) , 3–79.

[3] B. F. Feeney, R. J. Kappangantu, *On the Physical Interpretation of Proper Orthogonal Modes in Vibrations*, Sound Vibrations, 211, (1998) 607-616.

[4] R.A. Bialecki, A.J. Kassab, A. Fic, (2004), *Proper orthogonal decomposition and modal analysis for acceleration of transient fem thermal analysis*, International Journal for Numerical Methods in Engineering, 62, (2004), 774-797.

[5] A. Fic, R.A. Bialecki, A.J. Kassab, *Solving transient nonlinear heat conduction problems by proper orthogonal decomposition and the finite-element method*, Numerical Heat Transfer, 48, (2005), 103-124.

[6] Z. Ostrowski, R.A. Bialecki, A.J. Kassab, *Estimation of constant thermal conductivity by use of proper orthogonal decomposition*, Computational Mechanics, 37, (2005), 52-59.

[7] Z. Ostrowski, R.A. Bialecki, A.J. Kassab, *Solving inverse heat conduction problems using trained POD-RBF network inverse method*, Inverse Problems in Science and Engineering, 16, (1),(2008), 39-54.

[8] Z. Ostrowski, A. Klimanek, R.A. Bialecki, *CFD two-scale model of a wet natural draft cooling tower*, Numerical Heat Transfer, 57, (2010), 119-137.

[9] C. Rogers, A.J. Kassab, E. Divo, Z. Ostrowski, R.A. Bialecki, *An inverse POD-RBF network approach to parameter estimation in mechanics*, Inverse Problems in Science and Engineering, Vol. 20, No. 5-6, (2012), pp. 749-763.

[10] R.L. Hardy, *Multiquadric equations of topolography and other irregular surfaces*, Journal of Geophysical Research, 76, (1971), 1905–1915.

[11] R.L. Hardy, *Theory and applications of the multiquadric biharmonic method: 20 years of discovery 1968–1988*, Computational Mathematics with Applications, 19, (1990), (8–9):163–208. [6] Bejan A., and Almogbel M., Constructal T-shaped fins, *International Journal of Heat and Mass Transfer*, Vol. 43, 2000, pp. 2101-2115.