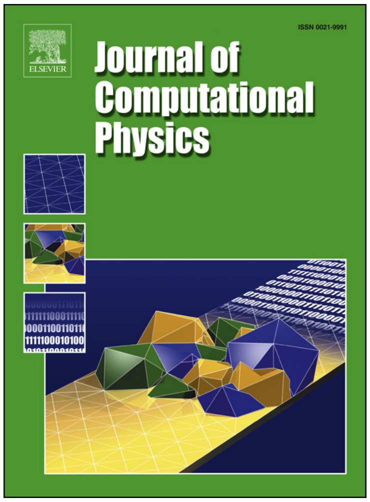
Accepted Manuscript

A parabolic velocity-decomposition method for wind turbines

Anshul Mittal, W. Roger Briley, Kidambi Sreenivas, Lafayette K. Taylor



PII: S0021-9991(16)30536-8
DOI: http://dx.doi.org/10.1016/j.jcp.2016.10.038
Reference: YJCPH 6916

To appear in: *Journal of Computational Physics*

Received date: 2 September 2015
Revised date: 18 August 2016
Accepted date: 13 October 2016

Please cite this article in press as: A. Mittal et al., A parabolic velocity-decomposition method for wind turbines, *J. Comput. Phys.* (2016), http://dx.doi.org/10.1016/j.jcp.2016.10.038

A Parabolic Velocity-Decomposition Method for Wind Turbines

Anshul Mittal[1], W. Roger Briley[2], Kidambi Sreenivas [3] and Lafayette K. Taylor[4]
SimCenter, University of Tennessee at Chattanooga, Chattanooga, Tennessee, 37403

An economical parabolized Navier-Stokes approximation for steady incompressible flow is combined with a compatible wind turbine model to simulate wind turbine flows, both upstream of the turbine and in downstream wake regions. The inviscid parabolizing approximation is based on a Helmholtz decomposition of the secondary velocity vector and physical order-of-magnitude estimates, rather than an axial pressure gradient approximation. The wind turbine is modeled by distributed source-term forces incorporating time-averaged aerodynamic forces generated by a blade-element momentum turbine model. A solution algorithm is given whose dependent variables are streamwise velocity, streamwise vorticity, and pressure, with secondary velocity determined by two-dimensional scalar and vector potentials. In addition to laminar and turbulent boundary-layer test cases, solutions for a streamwise vortex convection test problem are assessed by mesh refinement and comparison with Navier-Stokes solutions using the same grid. Computed results for a single turbine and a three-turbine array are presented using the NREL offshore 5-MW baseline wind turbine. These are also compared with an unsteady Reynolds-averaged Navier-Stokes solution computed with full rotor resolution. On balance, the agreement in turbine wake predictions for these test cases is very encouraging given the substantial differences in physical modeling fidelity and computer resources required.

Nomenclature

PNS	=	Parabolized Navier-Stokes
AL	=	Actuator Line
L	=	Reference length, length scale in the *x*-direction
D	=	Turbine diameter
u	=	*x*-direction velocity
ν	=	Kinematic viscosity
ρ	=	Reference density
Re	=	Reynolds number, uL/ν
p	=	Pressure
v, *w*	=	Velocities in *y* and *z* directions (transverse plane)
ϵ	=	Projection width, distance over which forces are distributed
$\vec{b}$	=	Body force per unit volume: $b_x\hat{i}+b_y\hat{j}+b_z\hat{k}$
h	=	Distance between boundary node and first point off the boundary

[1] Graduate Student, SimCenter: National Center for Computational Engineering, 701 East M. L. King Blvd. Chattanooga TN.

[2] Professor Emeritus, SimCenter: National Center for Computational Engineering, 701 East M. L. King Blvd. Chattanooga TN.

[3] Research Professor, SimCenter: National Center for Computational Engineering, 701 East M. L. King Blvd. Chattanooga TN.

[4] Professor (Retired), SimCenter: National Center for Computational Engineering, 701 East M. L. King Blvd. Chattanooga TN.

I. Introduction

A. Motivation

Modern computational simulation algorithms and supercomputing systems have become an increasingly powerful tool, providing detailed analysis of flow fields arising in practical engineering problems. As a result, computational methods play a significant role in design and engineering. Three-dimensional Navier-Stokes simulations are now able to provide very detailed flow field predictions for very large problems with considerable accuracy. However, despite their feasibility and broad applicability, many large high-fidelity Navier-Stokes simulations are too expensive or time consuming for solving practical engineering problems. For this reason, computational simulations based on approximate forms of the Navier-Stokes equations have been developed and used for many years.

Many approximate methods have been developed to reduce run times, especially for steady flow problems. A popular approach has been to devise approximations that yield equations solvable by a well-posed spatial-marching integration. These spatial marching approaches significantly reduce the computer resources required because steady-state solutions require only a single marching pass from the inflow boundary to the outflow boundary, a circumstance analogous to solving the boundary layer equations. A desktop computer is often sufficient rather than a supercomputer/cluster, and runtimes are reduced to minutes or hours instead of days or weeks. Spatial marching approaches have been researched for almost half a century, with different techniques emerging for incompressible, subsonic, supersonic, and interacted supersonic/subsonic problems. These diverse spatial marching approaches are often grouped under a general heading of Parabolized Navier-Stokes or PNS methods. Numerous PNS methods are reviewed in some detail in [1–3]. Interest in PNS methods still exists in various fields, mainly motivated by the high cost of Navier-Stokes simulations. The wind industry is an example, mainly because wind farm simulations are too costly without certain approximations, and a wide range of methods that vary in fidelity and cost are used.

Of particular interest here are wind turbine and wake simulations. Wind turbines are self-aligned with a relatively steady primary flow in the prevailing wind direction, and they cause a stream-tube expansion effect upstream of the turbine, and they generate complex wake flows downstream of the turbine. The flow near the wind turbine is both unsteady and highly three-dimensional due to the rotating blades. With modern supercomputers, Navier-Stokes methods are capable of providing high-fidelity simulations for flow through the detailed turbine geometry and in near and far wake regions, the primary limitation being turbulence modeling and the computing resources required. Time-averaged eddy-viscosity turbulence models are generally effective in thin shear layers on solid surfaces but are less well developed for turbine wake regions. Many Navier-Stokes simulations therefore employ large-eddy simulations (LES) that calculate large-scale unsteady turbulent structures without modeling, although at considerable computational expense. Although costly, recent rotor-resolved Navier-Stokes simulations are an effective tool for understanding wind turbine flows in detail. The Blade Element Momentum or BEM method is a simpler and widely used method for analyzing flow through a turbine and estimating its performance. It utilizes two-dimensional aerodynamic data for various airfoil sections (blade elements) in conjunction with a global momentum balance to estimate the aerodynamic forces acting on the blades. Other methods have included vortex-lattice and panel methods.

Another alternative is a hybrid approach using a Navier-Stokes solver to simulate the overall flow field near the turbine and in wakes, but with an embedded actuator model to approximate the turbine. Actuator models approximate the effect of the rotor on the flow in some average sense, without resolving the detailed rotor geometry. In increasing order of fidelity, the rotor can be modeled as an actuator disk (AD) [4], actuator line (AL) [5], actuator surface [6], or by full rotor resolution [7]. Actuator disk methods represent the turbine as a disk with aerodynamic forces averaged over the entire disk area. This approach models the gross effects of the turbine on the flow but omits individual rotor blades and blade tip vortices. Actuator line methods model individual rotor blades and predict local aerodynamic forces at each blade location to simulate the effect of individual blades rotating in the field. This unsteady model does capture physical features such as root and tip vortices.

A large number of models have been developed to simulate turbine wakes and their effects on downstream turbines. Simple kinematic models incorporate an algebraic expression for the wake velocity profile. Parabolic spatial marching methods (PNS) are used to generate flow fields in the far-wake region. Thorough reviews of these simulation methods and available tools are given by Vermeer et. al [8], Sanderse et. al [9] and Hansen et. al [10]. Higher fidelity methods applied to entire wind farms range from Navier-Stokes methods with embedded turbine models based on AD [11] or AL [12], to full rotor-resolved Navier-Stokes methods [13].

The present paper will present a low-cost steady mean flow model combining a parabolic spatial-marching method and a compatible turbine model. Together, these are capable of steady simulations of wind turbine flows, both upstream of the turbine and in downstream wake regions. The local effect of wind turbines and their upstream stream tube expansion is modeled as distributed source terms incorporating time-averaged aerodynamic forces generated by a BEM turbine model. The following sections will discuss this proposed methodology and its relation to previous PNS methods.

B. Parabolic Methods for the Far-Wake Region

One of the first wake models for application in wind energy was developed by Sforza et al. [14,15]. It is a linearized momentum equation with constant convective velocity and constant eddy viscosity. Two popular parabolic models have been developed by Ainslie [16,17] and Crespo et al. [18]. The Ainslie model assumes an axisymmetric wake and neglects the streamwise pressure gradients and the swirl in the wake. These assumptions are not suitable for the near-wake region, and so the solutions are initiated in the far-wake region by specifying a velocity deficit (wake characterization) at a location two or more rotor diameters downstream. The turbulent shear stresses are modeled using an eddy-viscosity closure scheme. This model does not account for ground effects or an atmospheric boundary layer.

The model developed by Crespo et al. is called the UPMWAKE model. The wind turbine is assumed to be immersed in a non-uniform surface layer described by analytical expressions that account for the effects of atmospheric stability and surface roughness. Streamwise viscous diffusion and axial pressure-gradient terms are neglected to obtain parabolic equations for the far-wake region. The system of equations is closed using a k-ε turbulence model and is solved numerically using the SIMPLE algorithm. This model also excludes the near-wake region and is initiated by specifying the velocity deficit created by the rotor at a location two or more rotor diameters downstream. Atmospheric stability effects are modeled using a Boussinesq approximation. The UPMWAKE methodology has been improved by Schepers et al. [19] and the resulting model is known as WAKEFARM. It uses a database of experimental velocity profiles to initialize the far-wake solution at a cross-section located 2.25 rotor diameters downstream of the wind turbine.

All of these far-wake models omit streamwise pressure gradients to obtain a parabolic approximation, and the far-wake solution is initialized at a certain distance downstream of the wind turbine. The required starting velocity profiles are thus important and must be chosen to model near-wake flow effects adequately. Schepers et al. [19] used a free-vortex method to calculate an approximation for streamwise pressure gradients in the far-wake region. This approximation is introduced by prescribing source terms to represent the pressure-gradient field within the framework of the WAKEFARM model.

C. Parabolic Velocity-Decomposition (PVD) Model

A new parabolic wind turbine model is presented here, based on the three-dimensional viscous primary/secondary flow approximation developed by Briley et al. [20,21]. This approximation employs a Helmholtz decomposition of the secondary flow velocity vector to isolate an irrotational transverse-velocity vector component analogous to the surface-normal velocity component of boundary-layer theory. This (small) vector is thereby separated from the rotational transverse velocity vector associated with streamwise vorticity and possibly (large) secondary flows. Guided by order-of-magnitude estimates, this small component vector is neglected in certain convection terms in the transverse momentum equations. Streamwise viscous diffusion terms are also neglected in the customary manner of PNS methods. These approximations produce a parabolic equation set for both viscous and inviscid cases, without any need to approximate the pressure-gradient terms.

The present approach qualifies as a PNS method since it solves a parabolic equation set by spatial marching. However, the method is distinct from the PNS methods analyzed in [1–3,16–19] because its inviscid parabolizing approximation is based on a velocity decomposition and physical order-of-magnitude estimates, rather than an axial pressure gradient approximation. For convenience, the method is referred to here as the parabolic velocity decomposition (PVD) method.

The PVD method is augmented by a new turbine model that allows the solution to be initialized well upstream of a turbine and to calculate the flow field by spatial marching through the turbine and continuing through the downstream wake regions. This is accomplished by treating the flow field as a steady flow in which changes in velocity, energy and vorticity caused by the turbine are modeled by a source-term distribution at the turbine and in the upstream stream tube expansion region. This distributed source-term model incorporates time-averaged aerodynamic forces predicted by the (FAST) BEM model developed at NREL [22]. This turbine model is analogous to a time-averaged actuator-line method implemented as a distributed actuator disk method. The PVD method and turbine model are applicable throughout the flow field, both upstream and downstream of the turbine.

The parabolizing flow approximation in the PVD model is based on the concept of a primary flow direction (straight or curved) whose velocity component does not reverse direction or encounter blunt surfaces with stagnation points. Significantly, it allows large streamwise vorticity and secondary-flow velocity and is well suited for both non-separating external flows past smooth surfaces and internal flows in smoothly curved ducts. This primary/secondary flow analysis has been assessed and validated previously for flow in ducts and 90-degree bends by comparison with both experimental measurements and Navier-Stokes solutions [20,21], as well as in turbofan-exhaust lobed-mixer geometries [23]. This methodology has been shown to be highly accurate when applied to flow problems in which the underlying assumptions of high Reynolds number, identifiable primary-flow direction, and no occurrence of reversed primary flow are satisfied. It is re-introduced here as a PNS-type approach for modeling wind turbines. In the case of a multi-turbine wind farm, the prevailing wind defines the primary-flow direction, and each turbine can be modeled locally using the distributed source-term model given here.

Basic validation of the methodology and software is addressed here in three test cases. The first two are simple laminar and turbulent flows past a flat plate. The third test case is a source-term induced, streamwise vortex convection problem for which the computed PVD solution is compared with a full Navier-Stokes solution. Finally, computed results are given for flow through a single and an array of NREL offshore 5-MW baseline wind turbines [24] at the design/rated wind speed of 11.4 m/s. The PVD solutions with turbine model are compared with the unsteady Reynolds-averaged Navier-Stokes (RANS) solutions computed with full rotor resolution.

In the remainder of this paper, the governing equations and solution methodology are discussed in Sections II and III, and details of the wind turbine model are given in Section IV. Computed results are discussed in Section V, and concluding remarks are given in Section VI.

II. Governing Equations

The nondimensional form of the Navier-Stokes continuity and momentum equations for steady and incompressible flow is given in Eq. (1). The dimensional reference quantities are L (x-direction length scale), u (x-direction velocity), ρu^2 (pressure), and ν (kinematic viscosity). The Reynolds number ($Re = uL/\nu$) is assumed to be large, and x is the primary flow direction.

$$\begin{gathered}\nabla \cdot \vec{U} = 0 \\ \vec{U} \cdot \nabla \vec{U} + \nabla p = \frac{1}{Re}\left(\nabla^2 \vec{U}\right) + \vec{b}\end{gathered} \tag{1}$$

The source term b is added here to accommodate the distributed source-term turbine model.

The PVD governing equations without the source terms were derived by Briley and McDonald [20] in a curvilinear coordinate system. This section summarizes the key definitions and equations including source terms and in Cartesian coordinates. The velocity vector $\vec{U} = \vec{U}_p + \vec{V}_s$ is written as the sum of primary and secondary flow velocities such that $\vec{U}_p = u\hat{i}$ and $\vec{V}_s = v\hat{j} + w\hat{k}$. A surface gradient operator is defined for the transverse y-z plane as $\nabla_s\left(\cdot\right) = \hat{j}\,\partial\left(\cdot\right)/\partial y + \hat{k}\,\partial\left(\cdot\right)/\partial z$. A Helmholtz vector decomposition is applied to the secondary-flow velocity such that

$$\begin{gathered}\vec{V}_s = \nabla_s \phi + \nabla_s \times \left(\hat{i}\psi\right) = \vec{V}_\phi + \vec{V}_\psi \\ \vec{V}_\phi = \hat{j} v_\phi + \hat{k} w_\phi = \hat{j}\frac{\partial \phi}{\partial y} + \hat{k}\frac{\partial \phi}{\partial z} \\ \vec{V}_\psi = \hat{j} v_\psi + \hat{k} w_\psi = \hat{j}\frac{\partial \psi}{\partial z} - \hat{k}\frac{\partial \psi}{\partial y}\end{gathered} \tag{2}$$

Without approximation, the continuity equation in these variables becomes

$$\frac{\partial u}{\partial x} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = 0$$

In the momentum equations, the primary/secondary flow approximations assume that $\left|\vec{V}_{\phi}\right| \square\ O(1)$ and that surface shear layers are thin, so that the following order-of-magnitude estimates are appropriate for high Reynolds number:

1) Throughout the flow field:

$$u,\ p,\ \partial(\cdot)/\partial x,\ v_{\psi},\ w_{\psi}\ =\ O(1)$$

$$v_{\phi},\ w_{\phi}\ =\ O\left(Re_x^{-1/2}\right)$$

2) In rotational inviscid regions: $\partial(\cdot)/\partial y,\ \partial(\cdot)/\partial z = O(1)$

3) In thin high-shear regions near solid surfaces whose surface-normal unit vector is $\vec{n}$:

$$\vec{n}\cdot\nabla = \frac{\partial(\cdot)}{\partial n} = O\left(Re_x^{1/2}\right), \quad \nabla - \frac{\partial(\cdot)}{\partial n}\cdot\vec{n} = O(1)$$

These scalings can be reduced to those of three-dimensional boundary-layer theory in which either y or z is the surface normal coordinate. Using these scalings, the momentum equations can be written as in Eqs. (3) below. Any viscous term containing $\partial/\partial x$ has been neglected based on these scalings. The streamwise vorticity is given by $\Omega_x = \hat{i}\cdot\left(\nabla\times\vec{U}\right) = -\nabla_s^2\psi$. The continuity equation remains unchanged.

$$\vec{U}\cdot\nabla u + \frac{\partial p}{\partial x} = \frac{1}{Re}\left(\frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) + b_x$$

$$\vec{U}\cdot\nabla v_{\psi} + \frac{\partial p}{\partial y} = \frac{1}{Re}\left(-\frac{\partial \Omega_x}{\partial z}\right) + b_y \tag{3}$$

$$\vec{U}\cdot\nabla w_{\psi} + \frac{\partial p}{\partial z} = \frac{1}{Re}\left(\frac{\partial \Omega_x}{\partial y}\right) + b_z$$

With this approximation the primary-flow momentum equation is unaltered, and the transverse momentum equations are modified only by neglecting gradients of the irrotational secondary-velocity components in the inviscid convection terms. Note that the irrotational secondary-velocity vector is determined by $\partial u/\partial x$ in the continuity equation, and the rotational secondary-velocity vector is determined by the streamwise vorticity Ω_x .

The characteristic analysis of these equations was given by Briley and McDonald [20] and is unaffected by the source terms added here. It shows that the governing equations are non-elliptic and well posed for spatial-marching solution, in both inviscid and viscous cases. These governing equations are reformulated for numerical solution by replacing the two transverse momentum equations by their surface divergence and surface curl. With some subsequent manipulation, the reformulated equations governing the five dependent variables $\Omega_x, \psi, p, u, \phi$ are given below.

$$\vec{U}\cdot\nabla\Omega_x + NL = \frac{1}{Re}\left(\frac{\partial^2\Omega_x}{\partial y^2}+\frac{\partial^2\Omega_x}{\partial z^2}\right)+\frac{\partial b_z}{\partial y}-\frac{\partial b_y}{\partial z}$$

$$\left(\frac{\partial^2\psi}{\partial y^2}+\frac{\partial^2\psi}{\partial z^2}\right)=-\Omega_x$$

$$\left(\frac{\partial^2 p}{\partial y^2}+\frac{\partial^2 p}{\partial z^2}\right)=-\frac{\partial}{\partial y}\left(\vec{U}\cdot\nabla\right)v_\psi-\frac{\partial}{\partial z}\left(\vec{U}\cdot\nabla\right)w_\psi+\frac{\partial b_y}{\partial y}+\frac{\partial b_z}{\partial z} \tag{4}$$

$$\vec{U}\cdot\nabla u+\frac{\partial p}{\partial x}=\frac{1}{Re}\left(\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+b_x$$

$$\left(\frac{\partial^2\phi}{\partial y^2}+\frac{\partial^2\phi}{\partial z^2}\right)=-\frac{\partial u}{\partial x}$$

where

$$NL=\left(\frac{\partial u}{\partial y}\frac{\partial w_\psi}{\partial x}+\frac{\partial v}{\partial y}\frac{\partial w_\psi}{\partial y}+\frac{\partial w}{\partial y}\frac{\partial w_\psi}{\partial z}\right)-\left(\frac{\partial u}{\partial z}\frac{\partial v_\psi}{\partial x}+\frac{\partial v}{\partial z}\frac{\partial v_\psi}{\partial y}+\frac{\partial w}{\partial z}\frac{\partial v_\psi}{\partial z}\right)$$

Note that the highest-order derivatives for each of these variables form a surface Laplacian operator in the cross-sectional plane.

III. Solution Methodology

The system of equations given in Eq. (4) is solved using a sequentially block-decoupled, semi-implicit algorithm. Letting $(\,\cdot\,)^n$ and $(\,\cdot\,)^{n+1}$ denote explicit and implicit levels of the marching discretization, an efficient semi-implicit algorithm is constructed by modifying the system of equations written as a fully-implicit backward-difference semi-discrete approximation. The primary modification is a common linearization of the convective operator as $\vec{U}^n\cdot\nabla(\,\cdot\,)^{n+1}$, where $\vec{U}^n=\vec{U}_p^n+\vec{V}_s^n$, and the second modification is an explicit evaluation of the nonlinear terms NL^n in the vorticity equation using values known from a previous step in the solution sequence. The result is the following semi-discrete system:

$$\vec{U}^n\cdot\nabla\Omega_x^{n+1}-\frac{1}{Re}\left(\frac{\partial^2\Omega_x^{n+1}}{\partial y^2}+\frac{\partial^2\Omega_x^{n+1}}{\partial z^2}\right)=\frac{\partial b_z^{n+1}}{\partial y}-\frac{\partial b_y^{n+1}}{\partial z}-NL^n$$

$$\left(\frac{\partial^2\psi^{n+1}}{\partial y^2}+\frac{\partial^2\psi^{n+1}}{\partial z^2}\right)+\Omega_x^{n+1}=0$$

$$\left(\frac{\partial^2 p^{n+1}}{\partial y^2}+\frac{\partial^2 p^{n+1}}{\partial z^2}\right)=-\frac{\partial}{\partial y}\left(\vec{U}^n\cdot\nabla\right)v_\psi^{n+1}-\frac{\partial}{\partial z}\left(\vec{U}^n\cdot\nabla\right)w_\psi^{n+1}+\frac{\partial b_y^{n+1}}{\partial y}+\frac{\partial b_z^{n+1}}{\partial z} \tag{5}$$

$$\vec{U}^n\cdot\nabla u^{n+1}-\frac{1}{Re}\left(\frac{\partial^2 u^{n+1}}{\partial y^2}+\frac{\partial^2 u^{n+1}}{\partial z^2}\right)=b_x^{n+1}-\frac{\partial p^{n+1}}{\partial x}$$

$$\left(\frac{\partial^2\phi^{n+1}}{\partial y^2}+\frac{\partial^2\phi^{n+1}}{\partial z^2}\right)=-\frac{\partial u^{n+1}}{\partial x}$$

Note that all of the equations are linear in the implicit unknowns $\Omega_x^{n+1},\psi^{n+1},p^{n+1},u^{n+1},\phi^{n+1}$. Furthermore, the highest-order terms in each of equations (Eq. (5)) is the surface Laplacian of one of these dependent variables, and these Laplacians fall on the diagonal of the system if the equations are ordered as in Eq. (5) above. Finally, without considering the boundary conditions, these equations would constitute a sequentially-decoupled system of equations in the ordered variable set $\left[\Omega_x^{n+1},\psi^{n+1},p^{n+1},u^{n+1},\phi^{n+1}\right]$. However, boundary conditions for velocity tangent to a boundary (no-slip condition) are usually required, and implicit treatment of this boundary condition linearly couples the Ω_x^{n+1} and ψ^{n+1} variables at the boundary. Consequently, the vorticity and vector potential equations are solved as a coupled 2×2 equation set. Equations (Eq. (5)) are thereby solved as a sequentially block-decoupled, second-order, linear system in the ordered variable set $\left[\Omega_x^{n+1},\psi^{n+1}\right]^T,\left[p^{n+1}\right],\left[u^{n+1}\right],\left[\phi^{n+1}\right]$. The diagram in Figure 1 illustrates the sequential block-decoupled structure of the equations and indicates the order in which the equations are solved. The dependent variables for each equation are shown inside the blocks.

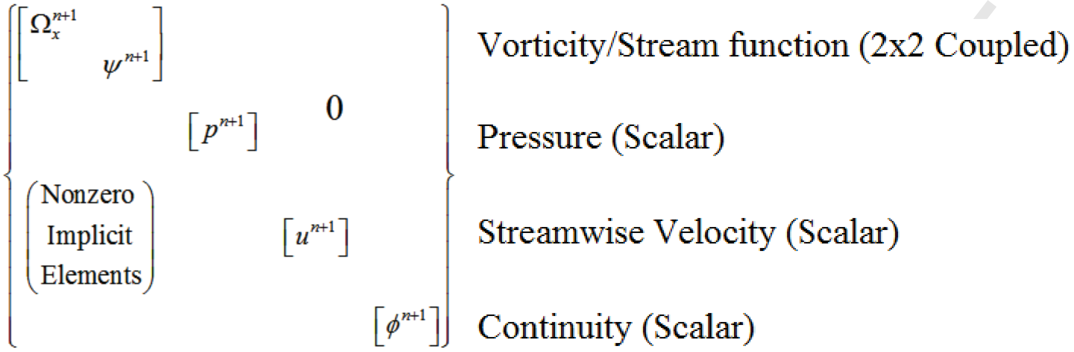


Figure 1: Diagram of sequential block-decoupling of the equations with their dependent variables

Given an initial or known solution at spatial step *n*, the solution at the next marching step, *n+1*, can be computed using a spatially discretized approximation of Eq. (5). First, the vorticity and the stream function equations are solved as a coupled linear system for $\Omega_x^{n+1},\psi^{n+1}$. Next the pressure equation is solved for p^{n+1} using secondary velocity components $\vec{V}_\psi^{\,n+1}$ evaluated from definitions in Eq. (2) with the just-computed stream function values ψ^{n+1}. The *x*-momentum equation is then solved for u^{n+1} using the updated pressure field p^{n+1}. Note that in the case of internal flows [20,21], the Neumann boundary conditions imply an arbitrary constant for p^{n+1}. For external flows, this is determined by the freestream pressure, and for internal flows, it can be evaluated by secant iteration to preserve the global mass flow rate. Finally, the continuity equation is solved for ϕ^{n+1} using updated values of u^{n+1}, and the secondary velocity components $\vec{V}_\phi^{\,n+1}$ are updated using ϕ^{n+1}.

A structured or unstructured grid can be used for the spatial discretization. The present study employed an unstructured finite-difference formulation, but all test cases employed Cartesian structured grids and were solved using the unstructured formulation and code. Derivatives in the transverse plane were computed using a weighted linear least-squares method [25,26].

The following expression is used to compute the derivative of a field variable *u* at a node *i* connected to *N* surrounding nodes denoted by j:

$$\frac{\partial u}{\partial x}_i = \sum_{e=1}^{N} W_e^x\left(\bar{u}_{j(e)}-\bar{u}_i\right) \tag{6}$$

The bars indicate a weighted quantity and in the present case, the weighting factor is the inverse of the distance between node *i* and node *j*. A similar expression can be written for the derivative in other spatial dimensions. A detailed derivation of the method and its efficient implementation are given by Hyams [25].

Streamwise derivatives were approximated using the standard 2-point backward-difference expression. The linear discretized 2×2 and scalar systems can be solved using any linear solver of choice. Lower-Upper/Symmetric Gauss-Seidel iteration was utilized in the present implementation. The solution is obtained by spatial marching in the *x*-direction from the inflow boundary to a specified outflow location.

A simple eddy-viscosity type turbulence model was used for turbulent flows. The streamwise vorticity and momentum equations are modified by incorporating nondimensional eddy and effective viscosities $\mu_{eff}=\mu_t+1$ and are given below for completeness:

$$\begin{aligned}\vec{U}^n \cdot \nabla\Omega_x^{n+1} - \frac{\mu_{eff}}{Re}\left(\frac{\partial^2\Omega_x^{n+1}}{\partial y^2}+\frac{\partial^2\Omega_x^{n+1}}{\partial z^2}\right) &= \frac{\partial b_z^{n+1}}{\partial y} - \frac{\partial b_y^{n+1}}{\partial z} - NL^n \\ \vec{U}^n \cdot \nabla u^{n+1} - \frac{\mu_{eff}}{Re}\left(\frac{\partial^2 u^{n+1}}{\partial y^2}+\frac{\partial^2 u^{n+1}}{\partial z^2}\right) &= b_x^{n+1} - \frac{\partial p^{n+1}}{\partial x}\end{aligned} \tag{7}$$

The key differences between the prior work of Briley and McDonald [20] and the present work are the 1) inclusion of the turbine model and associated source terms in the momentum equations, 2) use of a Cartesian coordinate system instead of curvilinear coordinates 3) using an unstructured finite-difference formulation in the transverse plane, 4) using LU/SGS iteration instead of the ADI method to solve the discretized linear system and 5) use of a turbulence model.

Given a specified initial solution at the inflow boundary, boundary conditions are required for all the boundaries in the transverse plane. Three different types of boundary conditions arise in the present solutions: farfield, viscous-wall, and symmetry conditions. The equations for these are given in Eq. (8) with $\hat{n}$ denoting the unit normal to the boundary and indices *w* and *i* denoting nodes on the boundary and first node off the boundary respectively.

$$\begin{array}{ccc} \text{Farfield} & \text{Viscous wall} & \text{Symmetry} \\ \begin{gathered}\Omega_x = \Omega_{x\infty} \\ p = p_\infty \\ u = u_\infty \\ \psi = \phi = 0\end{gathered} & \begin{gathered} u = \psi = 0 \\ \frac{\partial\phi}{\partial n} = 0 \\ \hat{n} = \hat{j} : \frac{\partial p}{\partial y} = \frac{\partial\Omega_x}{\partial z} \; and \; \Omega_{xw} + \frac{2}{h^2}\psi_i = \frac{2}{h}\frac{\partial\phi}{\partial z} \\ \hat{n} = \hat{k} : \frac{\partial p}{\partial z} = -\frac{\partial\Omega_x}{\partial y} \; and \; \Omega_{xw} + \frac{2}{h^2}\psi_i = -\frac{2}{h}\frac{\partial\phi}{\partial y}\end{gathered} & \begin{gathered}\frac{\partial\Omega_x}{\partial n} = \frac{\partial p}{\partial n} = \frac{\partial u}{\partial n} = \frac{\partial\phi}{\partial n} = 0 \\ \psi = 0\end{gathered} \end{array} \tag{8}$$

IV. Wind Turbine Model

The PVD method requires a steady-flow model of the turbine. The present turbine model consists of a field of distributed force vectors, specified as source terms in a cylindrical volume upstream of the turbine. This source-term distribution is constructed from forces predicted by a separate blade-element/momentum (BEM) analysis of the turbine geometry, in this case obtained from the open-source code FAST [22] from NREL. The FAST code simulates a wind turbine using a time-marching algorithm to predict radial distributions of the force vector on the rotating blades at each time step. A modified version of FAST available through the SOWFA [27] suite from NREL provides a software API for transfer of velocity and force data, along with blade position. This API is utilized by the PVD code to transfer velocities to and receive forces from the FAST code. The spatial distribution of time-averaged disk forces is generated using the procedures given below.

A. Gaussian distribution for approximate axial extent of upstream effects

The unsteady FAST BEM model defines force-vector distributions along rotating lines representing rotor blades, as a function of time. These forces from the FAST code are first time-averaged to yield a steady spatial distribution of the force vector in a disk of diameter D in the rotor plane, analogous to an actuator disk representing the turbine. The time-averaged forces on the disk are then used to calculate an axial distribution of source terms in a cylindrical volume upstream of the turbine. The cylindrical volume begins at an axial location about one rotor diameter upstream of the rotor disk, and ends at the rotor disk. This axial distribution is intended to model the deceleration and stream-tube expansion effect of the turbine on the upstream flow, which is otherwise not present in the parabolic model. This axial length, L_U, is an *a priori* modeling parameter representing the physical extent of upstream influence of the turbine. This length is known to be of order one rotor diameter, which is consistent with computed results from the Navier-Stokes test case to be given here, as well as superficially unrelated flows such as flow in a straight duct approaching a curved bend.

The source-term distribution along each axial line in the cylindrical volume is calculated from a one-dimensional Gaussian distribution function based on the corresponding value in the disk of time-averaged BEM forces. The Gaussian distribution, as shown in Eq. (9), gives the force for a node that is *x* distance away from the turbine disk in the streamwise/axial direction. The force application is done in a conservative fashion such that the sum of the forces applied at all the planes for a node is equal to the time-averaged BEM forces computed for that node. This is done using the scaling factor α . The expression for the scaling factor involves summation over all the planes in the cylindrical volume region i.e., from one rotor diameter upstream of the rotor disk to the rotor disk itself ($L_U = D$). The number of planes in this region depends on the spatial marching step chosen.

$$\begin{aligned} \vec{f}_p &= \frac{\vec{F}}{\alpha} \exp\left[-\left(x/L_U \right)^2 \right] \\ \alpha &= \sum \exp\left[-\left(x/L_U \right)^2 \right] \end{aligned} \tag{9}$$

B. Gaussian in-plane distribution

Implementation of the turbine model must also allow the cross-sectional grid used in the PVD solution to differ from the polar-coordinate grid containing the time-averaged BEM forces. A convenient method of transferring the BEM forces from the polar grid to the PVD grid is to use a separate Gaussian distribution applied in each cross sectional plane. A (nonphysical) projection-width parameter ϵ in this cross sectional Gaussian distribution controls the size of the region over which the BEM forces are transferred to the neighboring PVD grid points.

The forces computed by FAST are time averaged and stored on a polar grid with 36 circumferential points and 17 radial points (total of 612 points). The number of circumferential points is chosen to ensure sufficient resolution in the circumferential direction. The number of radial points is equal to the number of points used to discretize the blade in FAST. Since the polar grid points typically do not coincide with the PVD computational grid, forces for the computational grid are transferred (from polar grid to PVD grid) by assuming a Gaussian distribution as in Eq. (10).

Here r_{ip} is the distance between a computational node, *i*, and a polar grid point, *p*. This distance is nondimensionalized by the projection width, ϵ . The force transfer is done in a conservative fashion by utilizing the scaling factor β . The scaling factor is computed for every point on the polar grid. This ensures that the force at every polar grid point is distributed on to the PVD grid in the exact amount.

$$\begin{aligned} \vec{f}_i &= \sum_p \frac{\vec{f}_p}{\beta_p} \exp\left[-\left(r_{ip}/\epsilon \right)^2 \right] \\ \beta_p &= \sum_i \exp\left[-\left(r_{ip}/\epsilon \right)^2 \right] \end{aligned} \tag{10}$$

The wind turbine model described above in section A is derived so that the upstream influence of the turbine on the flow can be modeled, which is otherwise not possible in parabolic models. The second part of the model (section B) is for distribution of the forces from one grid in a plane to another grid in the same plane. This actuator-disk type model is not sensitive to the choice of projection-width parameter, ϵ because the distribution is between two grids that span a similar amount of area in a plane. The projection width was calculated as $\epsilon = 0.5\pi D/\text{Number of circumferential points in the polar grid}$, which for the present study is $\epsilon = 0.5\pi D/36$. This is unlike actuator-line models that transfer BEM data to a Navier-Stokes code without modeling upstream influence and without time averaging to remove rotating-blade unsteadiness. These Navier-Stokes actuator line models have been observed to be sensitive to the projection-width parameter for non-dimensionalizing distance.

C. Simulation Process

At each spatial marching step, the distance to each downstream wind turbine is computed, and if it crosses a threshold value of one rotor diameter, the turbine model is used to calculate source terms associated with that particular turbine as follows: The velocity field from the previous plane (see Figure 2) is used as input to run FAST for a total of 4 revolutions of the turbine. The induction feature is enabled in FAST to simulate the slowing down of the flow as it approaches the turbine to obtain accurate aerodynamic forces, and the tip-loss and hub-loss models are also switched-on in FAST to account for these losses. The forces computed by FAST are then time averaged over the last 2 revolutions to neglect/discount the effect of start-up transients. These time-averaged forces are stored on a polar grid as described above. The FAST program is called only once for each turbine encountered, and the resulting turbine model forces are stored for subsequent use as the PVD simulation progresses.

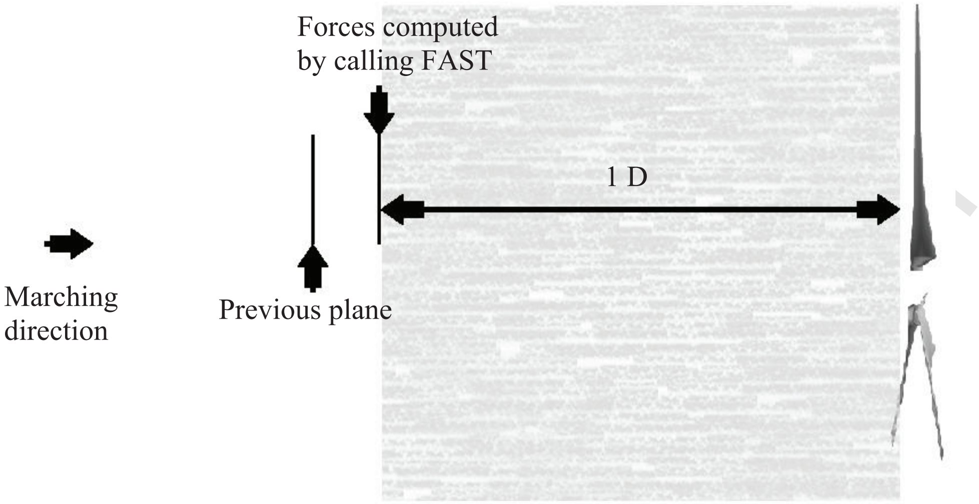


Figure 2: Illustration showing the location where FAST is called. Forces are distributed over the shaded region

To approximate the upstream-influence effect of the turbine, the forces are distributed among all planes from one rotor diameter upstream of the turbine to the turbine itself using a Gaussian distribution (shown by the shaded region in Figure 2). At each plane, Gaussian in-plane calculation is carried out to transfer the forces from the polar grid to the PVD computational grid as discussed above. It should be noted that in the present model, the pressure field does not react to the streamwise source term, which is unimportant because the streamwise source term itself acts as a streamwise pressure gradient that models the upstream deceleration, resulting in a physically correct velocity field. However, the effect of the secondary velocities and the transverse source terms on the pressure field is captured.

V. Results

The present PVD model is validated and verified using three different test cases. First two test cases are basic verification that the method correctly simulates flow in laminar and turbulent flat plate boundary layers, whose solutions are well known. The third test case is axial convection of a source-term induced streamwise vortex, and computed results are compared with Navier-Stokes solutions. The PVD model is used to simulate the flow through the NREL 5 MW rotor [24], and this computed solution is compared with Reynolds-averaged Navier-Stokes (RANS) solutions with fully resolved rotor geometry. As a practical application example, flow through an array of three turbines is also simulated, and PVD results are compared with the blade-resolved RANS solutions.

A. Laminar flow over a flat plate

Laminar flow over a flat plate is simulated for a unit Reynolds number of 20,000 per meter. The step size is chosen as 0.01 m for marching in the streamwise direction. The first point off the wall is 10^{-3} m and the mesh contains 671 nodes. To avoid the singularity at the leading edge of the plate, the solution (velocity profile) is initialized at planes corresponding to x=0.01 m and x=0.02 m. The boundary layer thickness (for initializing the solution) is computed using Eq. (11) and only the streamwise velocity is specified using Eq. (12). Note that the velocity normal to the wall/plate (y-velocity) is initialized to zero at both the planes.

$$\delta = 4.91x / \sqrt{\mathrm{Re}_x} \tag{11}$$

$$u = 1.5\left(\frac{y}{\delta}\right) - 0.5\left(\frac{y}{\delta}\right)^3 \tag{12}$$

The computations were conducted for a total of 200 steps and the final solution at x=2 m is shown along with the Blasius solution. Figure 3 shows a plot of the nondimensional x-velocity against the similarity variable ($\eta = y\sqrt{u_\infty/2\nu x}$) on the left axis whereas error in the computed solution is plotted on right axis. The maximum error in the solution is less than 0.7 % at around $\eta = 3$. The agreement, as can be seen, is good.

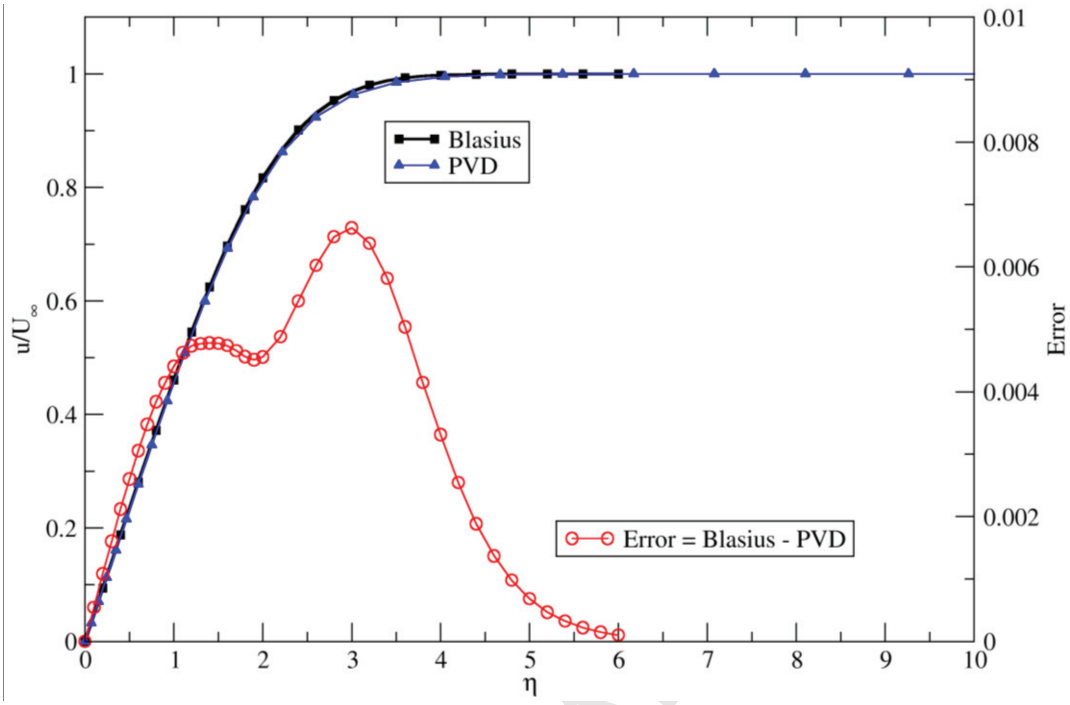


Figure 3: Plot of *x*-velocity against the similarity variable for laminar flow over a flat plate

B. Turbulent flow over a flat plate

For turbulent closure, the Spalart-Allmaras [28] model is utilized. The turbulence model is updated after each marching step of the mean-flow (PVD) equations. The turbulence model provides an eddy-viscosity field that is added to the molecular viscosity in the streamwise vorticity and momentum equations. The unit Reynolds number for the flow is 10^6 per meter. The total number of nodes is 793 and the off-the-wall spacing is 10^{-5}. The solution is marched for 100 steps (step size of 0.01 m). However, the marching is started at *x*=5 m from the leading edge. This was done to avoid the transition region and ensure that the flow is fully turbulent. The solution is initialized at planes corresponding to *x*=5.01 m and *x*=5.02 m with the Coles' law of the wake (Eq. (13)) [29]. The value of the Coles' parameter, Π, is 0.45 and boundary layer thickness , δ, is assumed to be 0.06 m. The values of von Karman constant and *B* are 0.41 and 5.0 respectively.

$$u^+ = \frac{1}{\kappa}\ln\left(y^+\right) + B + \frac{2\Pi}{\kappa}\sin^2\left(\frac{\pi y}{2\delta}\right) \tag{13}$$

The initialized solution at *x*=5.01 m and the computed solution at *x*=6 m are plotted against the law of the wall in Figure 4. The velocity profiles obtained from various locations on the plate were found to be self-similar. The skin friction coefficient computed at the end of the computations is $2.68\cdot10^{-3}$. The value of skin friction coefficient calculated using Eqs. (14) [29] and (15) [30] is $2.90\cdot10^{-3}$ and $2.61\cdot10^{-3}$ respectively. The agreement of the final PVD solution is excellent. The boundary layer thickness is estimated to be about 0.068 m at *x*=6 m.

$$C_f = 0.027\,\mathrm{Re}_x^{-1/7} \tag{14}$$

$$C_f = 0.0592\,\mathrm{Re}_x^{-1/5} \tag{15}$$

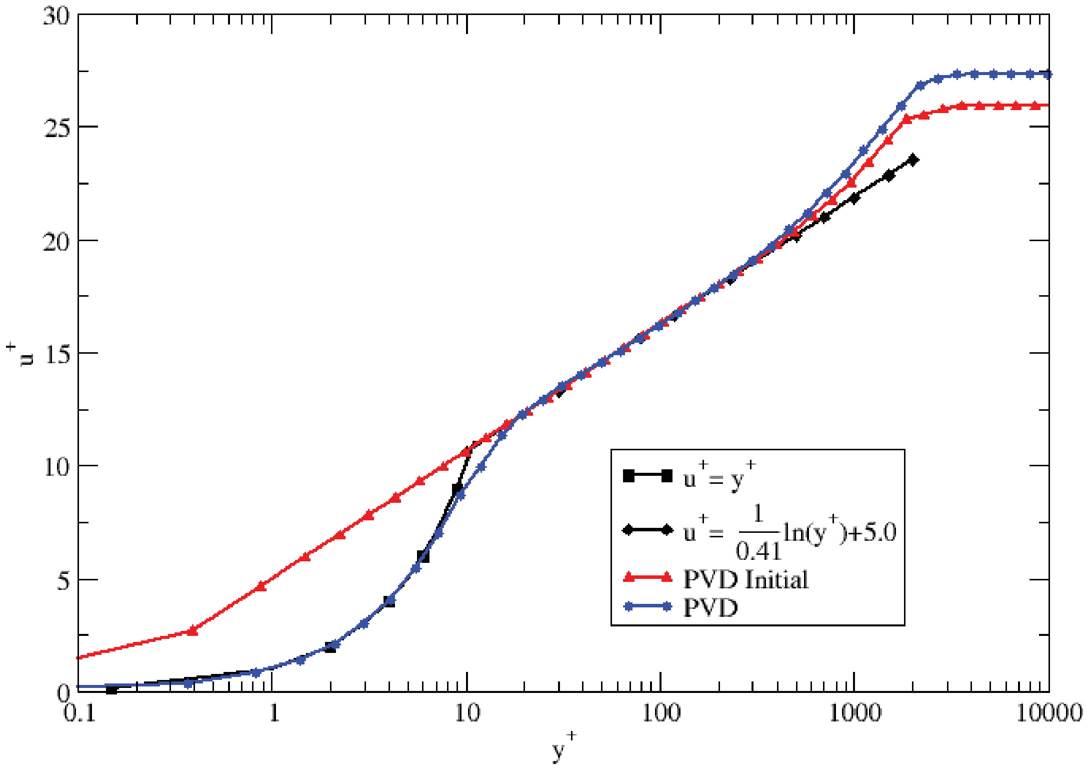


Figure 4: Plot of *x*-velocity for the turbulent flow over a flat plate

C. Axial convection of a streamwise vortex

In the previous two test cases, the streamwise vorticity and the stream function are identically zero in the entire flow region. This test case is designed to test the methods for a case in which a streamwise vortex is generated by forces applied as source terms. The vortex is generated by imposing forces in the transverse plane (*y*-*z* plane) near the start of the solution. The vortex is convected downstream (+*x*) by a uniform axial flow. The grid for this case extends 1 m in the *y* and *z* directions and has 41 points (uniformly spaced) in both directions. The center of the vortex is at (0.5, 0.5) with a radius of 0.1 m. The force per unit volume applied for creating the vortex is 12.25 N/m^3 (0.1 nondimensional units) in the tangential direction as given by Eq. (16). The marching step size is 0.01 m and the forces are applied on five consecutive planes from 0.03 m to 0.07 m in the marching direction. The Reynolds number for the flow is $6.8 \cdot 10^5$ and the flow is assumed to be laminar.

$$F_{tangential} = 12.25 \qquad 0 \le r \le 0.1 \tag{16}$$

To assess the accuracy of the computed results, the same flow conditions are simulated using a Navier-Stokes code, applying the same source-term forces to generate the vortex. The grid for this Navier-Stokes simulation is generated by extruding the planar PVD grid with the spatial marching step size (0.01 m). This ensured that the spacing in all the three directions (Δx , Δy and Δz) is exactly the same for both simulations. The Navier-Stokes simulation was run as a steady-state problem.

Table 1 compares the PVD and Navier-Stokes solutions, computed with identical grids. The Navier-Stokes results are presented for three schemes having second-order, fifth-order and seventh-order spatial accuracy. The Table also includes an additional seventh-order Navier-Stokes solution was also computed with the same streamwise step size but with a finer (81x81) cross sectional grid instead of the original (41x41). The maximum and minimum values of *y*-velocity and streamwise vorticity at three downstream locations are presented in the table. It is first observed that the vortex strength is dissipated in the Navier-Stokes solutions as it is convected downstream, whereas the PVD solution has minimal dissipation. This same trend is also evident in Figure 5, which plots the *y*-velocity profile along a horizontal line from outside the vortex to its center, at three different downstream locations. The plot compares the PVD solution with the seventh-order Navier-Stokes solution. Both Table 1 and Figure 5 show that the dissipation is minimal for the PVD solution. The amount of dissipation in the Navier-Stokes solutions is large for the second order scheme and significantly smaller for the seventh order scheme.

It is noted that streamwise vortex dissipation should be minimal for a laminar flow at this high Reynolds number, and in these results, the PVD simulation preserves the streamwise vortex with little dissipation. This difference in vortex dissipation between PVD and Navier-Stokes algorithms is possibly related to the numerical formulation in Eq. (4) in terms of the solution variables $\Omega_x, \psi, p, u, \phi$. This formulation differs from that of the Navier-Stokes in that it solves directly for streamwise vorticity transport, and the cross sectional Laplacian operator governs the highest order derivatives of each variable. Velocity is the fundamental solution variable in the Navier-Stokes equations, and vorticity is obtained by numerical differentiation of velocity. In fact, two-dimensional vorticity/stream-function formulations were popular in the past but were found to be inconvenient in three dimensions.

The PVD and Navier-Stokes velocity profiles in Figure 5 are in reasonable agreement, but there is a slight nonphysical overshoot in the PVD solution near the 0.35 m location. This is believed to be the result of the particular second-derivative approximation used for Laplacian terms. The PVD scheme used a weighted linear least-squares approximation for first derivatives that is commonly used for unstructured grids, even though a Cartesian grid was used for these test calculations. Second derivatives were computed by differentiating this first derivative using the same least-squares approximation, which is less accurate than the usual Cartesian central difference approximation. We believe this is the cause of the overshoot in this case, since it has not occurred in other calculations using the centered approximation. It is also observed that the velocity at the center of the vortex does not remain zero. This is seen for both the PVD and Navier-Stokes solutions.

To further investigate, these two observations (nonphysical overshoot and nonzero velocity at the center), PVD simulations were run on successively finer grids with number of grid points in the *y* and *z* directions ranging from 41 to 321. The spatial marching step size was held constant because the cross sectional resolution is important due to the discontinuity in applied force, whereas the solution is insensitive to axial step size due since the vortex does not change much after being generated. The results at 1 m downstream location are plotted in Figure 6. The nonphysical overshoot is present in all the simulations, however, the location of the overshoot has moved. As the grid is refined, the location of the overshoot moves closer to the edge of the vortex, which is at 0.4 m. On the finest grid, the magnitude of the nonzero velocity at the center of the vortex has reduced slightly.

Finally, it is observed that the fine-grid Navier-Stokes solution in Table 1 agrees with the PVD velocity but the streamwise vorticity is significantly larger. This is caused by the discontinuous force distribution applied to generate the vortex for this test case. The magnitude of the force is exactly zero at the center of the vortex and immediately outside the vortex radius. As a result, even though the velocity field is similar for the coarse and fine grids, the gradients of the same velocity field (and hence vorticity) are larger for the fine grid due to the spacing differences. This was confirmed by conducting both Navier-Stokes and PVD simulations on successively finer grids.

Although no experimental or theoretical data are available for comparison, these results give a useful comparison of PVD and Navier-Stokes simulations for an identical problem and grid with a vortex generated by source terms. These results provide at least some evidence that the PVD solution algorithm has good vortex preservation properties.

Table 1: Comparison of PVD and Navier-Stokes solutions for the vortex convection case

	2nd order spatially accurate scheme		5th order spatially accurate scheme		7th order spatially accurate scheme		7th order spatially accurate scheme (finer grid)		PVD	
Location (m)	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.
					y-velocity $\left(\times 10^{-2}\right)$					
0.2	5.184	-5.313	5.372	-5.603	5.271	-5.698	5.162	-5.359	5.584	-5.695
1.0	3.994	-3.995	4.901	-5.031	5.135	-5.466	5.248	-5.262	5.585	-5.692
1.8	3.336	-3.318	4.484	-4.553	4.884	-5.116	5.255	-5.357	5.595	-5.692
					Streamwise vorticity					
0.2	0.8014	-2.948	0.8011	-3.104	0.8214	-3.124	1.810	-6.030	1.109	-3.999
1.0	0.7460	-2.162	0.7090	-2.818	0.6874	-3.051	1.594	-5.315	1.110	-3.993

1.8	0.6604	-1.780	0.7324	-2.561	0.6970	-2.949	1.449	-4.699	1.109	-3.987

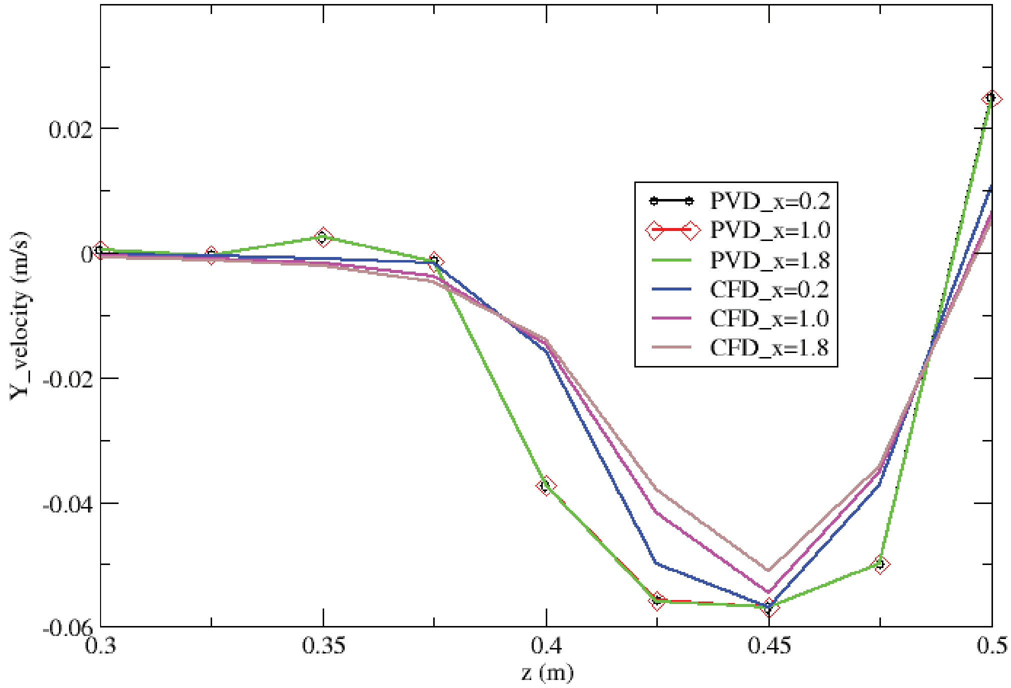


Figure 5: Comparison of *y*-velocity for the PVD and Navier-Stokes solutions (7th order) at different downstream locations

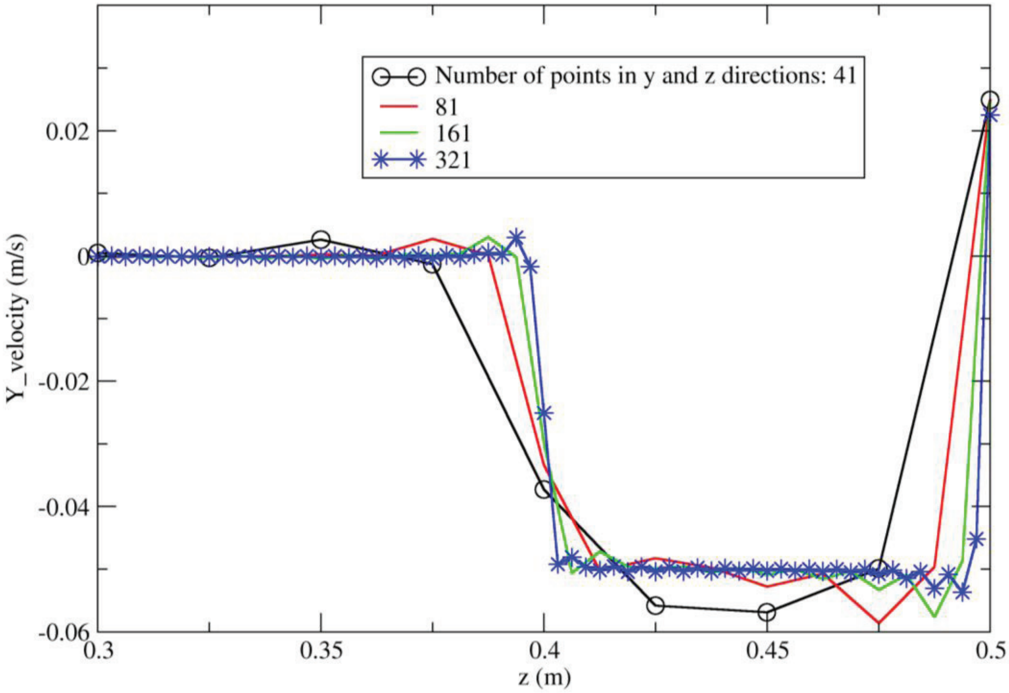


Figure 6: PVD solution 1 m downstream of the vortex on successively finer grids

D. Wind turbine

The final test case is flow through a NREL offshore 5-MW baseline wind turbine [24]. The turbine diameter is 126 m, and the simulation is conducted at the design point with a uniform inflow wind speed of 11.4 m/s. The grid extends to 500 m in *y* and *z* directions from the middle of the plane where the wind turbine is assumed located. Thus, the boundaries are approximately four rotor diameters away from the wind turbine center. The grid contains 10,201 nodes with spacing between the nodes ranging from 40 m at the boundaries to 4.05 m near the center. The spatial marching step size is 5 m and the simulation is run for 200 steps (1000 m). The turbine is assumed to be 150 m from the inflow boundary.

The PVD simulation for this test case used the standard Spalart-Allmaras (SA) turbulence model, without any special modifications. Although the SA and other eddy-viscosity models are generally successful in modeling turbulence in thin shear layers on solid surfaces, they are less well developed for turbulence encountered in wind turbine wakes. Since the PVD simulation does not model the actual blade surfaces, the near wake is caused entirely by the distributed BEM forces, and the tip vortices are averaged into a circumferential vortex sheet. The turbulence model then produces downstream turbulence from gradients in the axial and circumferential velocity, which is expected to be minimal. The turbulence generated in shear layers on individual blade surfaces is not directly modeled, although it is included in blade-resolved RANS simulations using the same turbulence model.

For comparison, a higher resolution unsteady RANS simulation of the same wind turbine was conducted at this same design point. The RANS simulation utilized the same Spalart-Allmaras turbulence model, but with full resolution of the rotor geometry, which requires a much finer grid than the PVD solution. The PVD grid near the center of the turbine had a spacing of 4.05 m whereas the RANS grid had a spacing of 1.25 m in the wake region. The fine spacing in the RANS grid was needed to maintain solution accuracy with low numerical wake dissipation, whereas the choice of PVD grid spacing was influenced by the relatively coarse blade discretization used in the BEM model. This test case will highlight differences between the moderate-fidelity, economical, coarse-grid PVD model and the higher-fidelity, blade-resolved, high-resolution RANS model. In comparing the PVD and RANS solutions, it should be kept in mind that they have the same turbulence model, but the physical models are distinctly different, and the solutions have different solution algorithms and grid resolution. Recall that the previous test results for convection of a force-induced vortex showed that the PVD model had less streamwise vortex dissipation than the RANS model with identical grids and applied forces.

Table 2 compares the torque and thrust values obtained for the PVD and unsteady RANS models, along with results reported by NREL [24]. The forces and the implied torque and thrust values are computed by the BEM code FAST and serve as an input to the PVD model. The 3-7% difference in the thrust and torque between PVD and NREL results is likely related to the different number of blade points in the discretizations. The 3-4% difference between PVD and RANS seems reasonable in view of the large differences in physical modeling and resolution.

The PVD solution is further assessed by comparison with time-averaged results from the RANS solution in the wake region downstream of the turbine. Figure 7 gives contour plots of axial (streamwise) velocity distributions in an axial plane passing through the center of the turbine from one diameter upstream of the turbine to six diameters downstream of the turbine. The white lines show the location of the turbine rotor, two diameters downstream location in the wake and five diameters location in the wake with the height of the line being the turbine diameter. The grid for both the cases is also shown in the figure. First, consider the wake development in the near-wake region immediately downstream of the wind turbine rotor. The RANS solution represents the time-averaged behavior of the unsteady solution with fully resolved rotor blade, hub, and tip geometries. The gradual near-wake expansion in the RANS solution reflects its high-fidelity treatment of rotating blade geometries, including tip vortex effects and modeling of turbulence produced on blade surfaces. The lower cost PVD model has no rotor-blade resolution and is only expected to approximate the turbine and near-wake initiation by incorporating time-averaged blade forces from the BEM model. The near wake in the PVD solution is established entirely by the distributed time-averaged BEM forces. The same turbulence model produces turbulence from gradients in the axial and circumferential velocity.

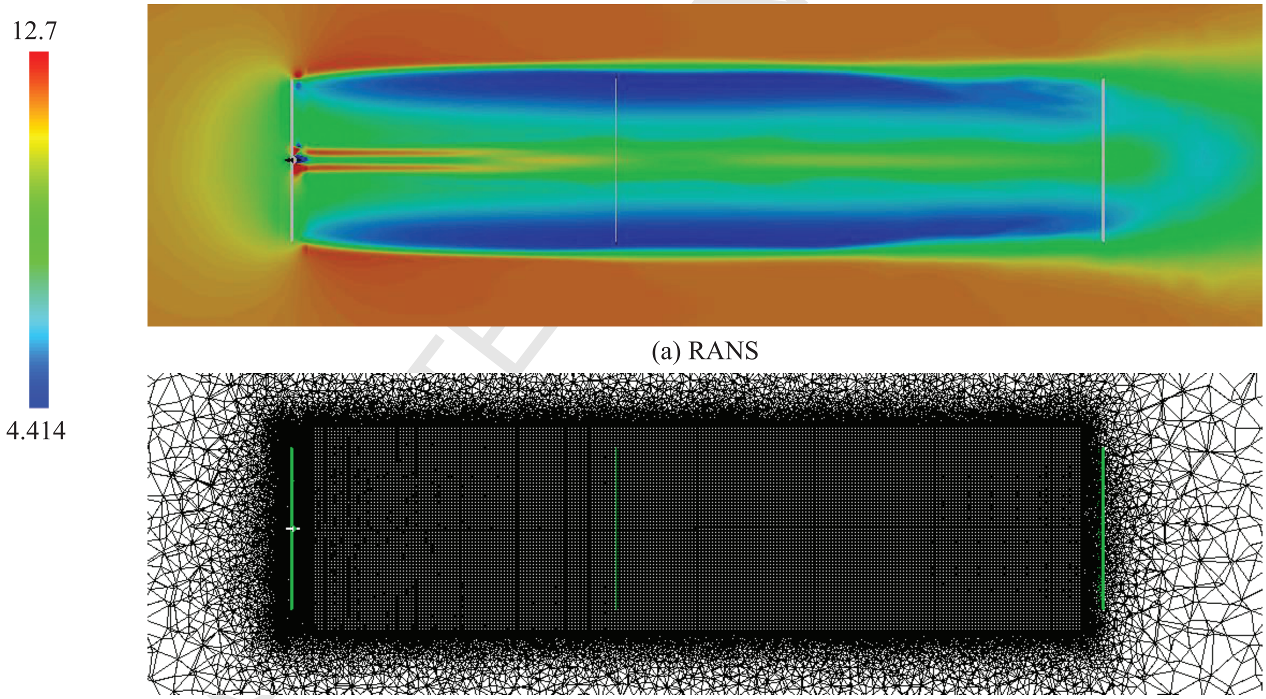
Beyond the near wake region (about one and a half rotor diameter downstream), the RANS wake is stable and does not dissipate or expand over the next two diameters. It is important to note that the PVD model does predict a wake whose diameter and cross-sectional velocity distribution agree reasonably well with the RANS solution in this intermediate wake region. Further downstream, the RANS wake expands rather suddenly. This expansion is attributed to numerical dissipation rather than physical behavior. The RANS grid was generated to provide very high resolution in the rotor and near-wake regions, but the grid is coarse after five rotor diameters downstream of the turbine, and this coarse grid spacing causes numerical dissipation. The PVD solution used a uniform Cartesian grid and has little downstream wake expansion. This lack of downstream expansion could indicate that the turbulence is inadequately modeled, but it could also imply less numerical vortex dissipation, as suggested in the previous test case. Neither of these solutions accounts for ambient turbulence in the surroundings.

Figure 8 compares axial velocity contours for the RANS and PVD solutions at a plane located two rotor diameters downstream of the turbine. The PVD velocity is higher at the center of the wake due to the absence of the hub. Moreover, since only 36 circumferential stations were used for storing the forces, a large Gaussian projection width was needed to ensure a reasonably smooth flow field. These two factors contribute to the weaker PVD wake seen in the figure, but otherwise the two wakes are in good agreement. Streamwise velocity is plotted at the same plane across the horizontal line (*z*-axis) passing through the center in Figure 9. The plot confirms that the velocity is higher at the center of the wake for PVD solution by about 0.5 m/s and the weaker wake, particularly, near the edge of the wake by about 0.5 m/s. The wake is smeared in the PVD solution due to the usage of a large Gaussian projection width as discussed above.

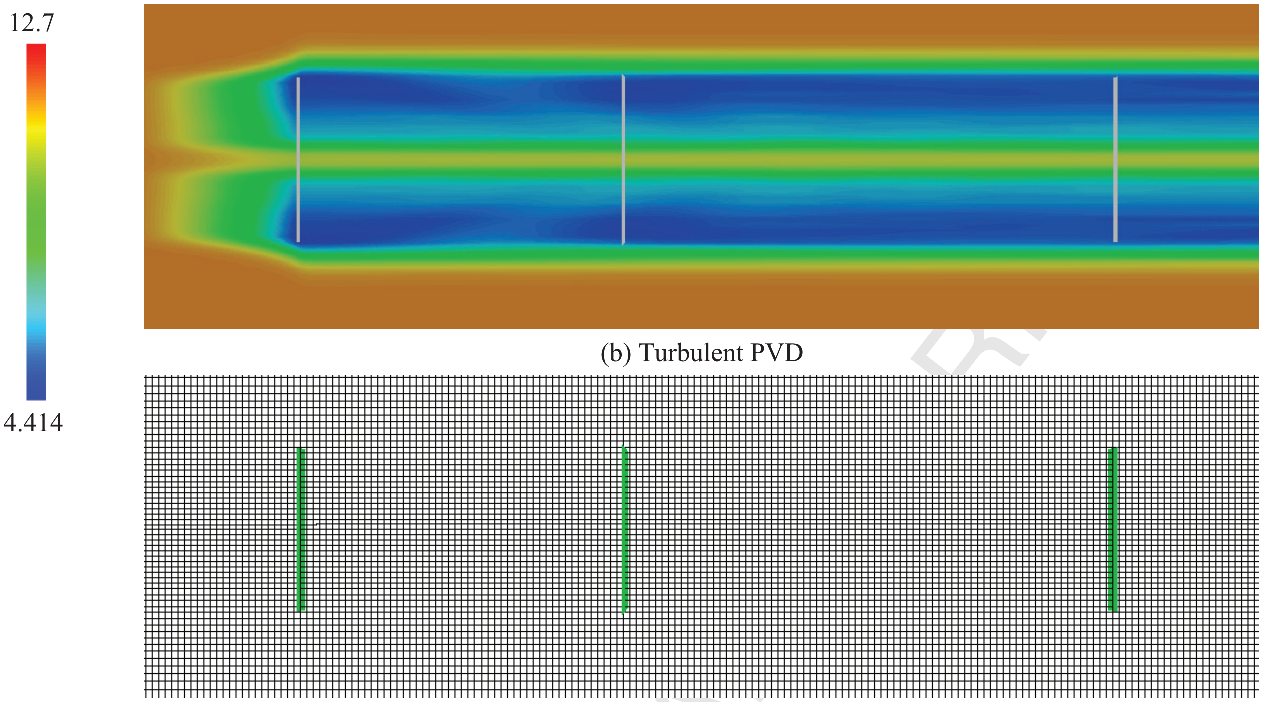
Figure 10 compares the secondary velocity component (*y*-velocity) between the RANS and PVD solutions at the same plane two diameters downstream of the turbine. Good comparison in the overall structure and the rotational velocity of the wake is observed in the figure between the RANS and PVD solutions.

Table 2: Comparison of thrust and torque values obtained using different simulation methods

	Reported by NREL	RANS simulation	PVD simulation
Thrust (kN)	800	762	741
Torque (kNm)	4180	4131	4311

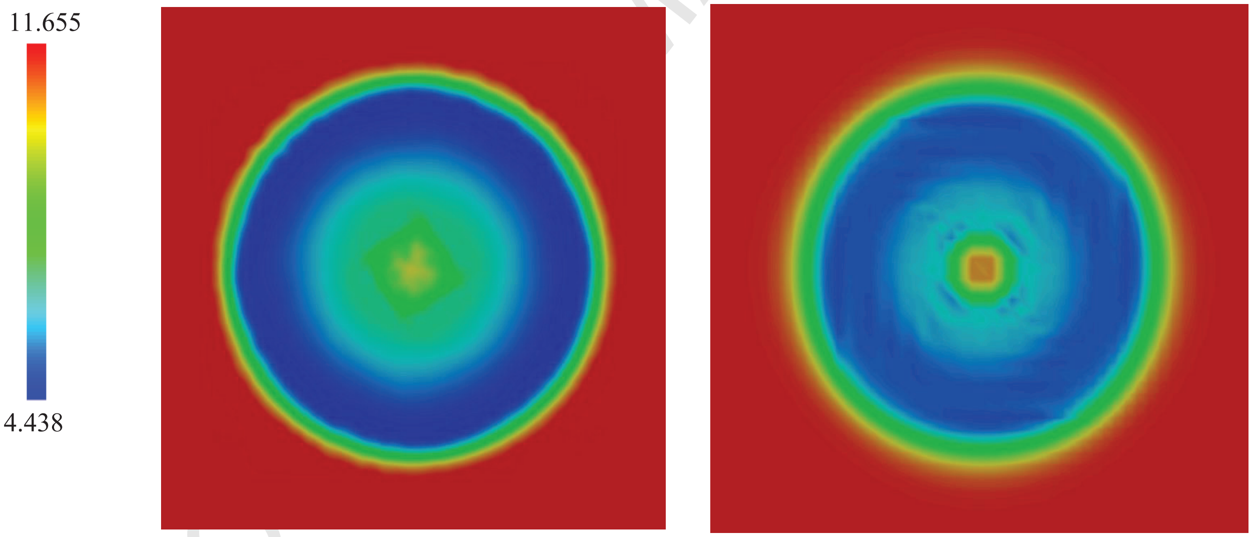


(a) RANS



(b) Turbulent PVD

Figure 7: Contours of axial velocity comparing the wake between RANS and PVD solutions



RANS

PVD

Figure 8: Contours of streamwise (*x*-velocity) velocity at two rotor diameters downstream of the wind turbine

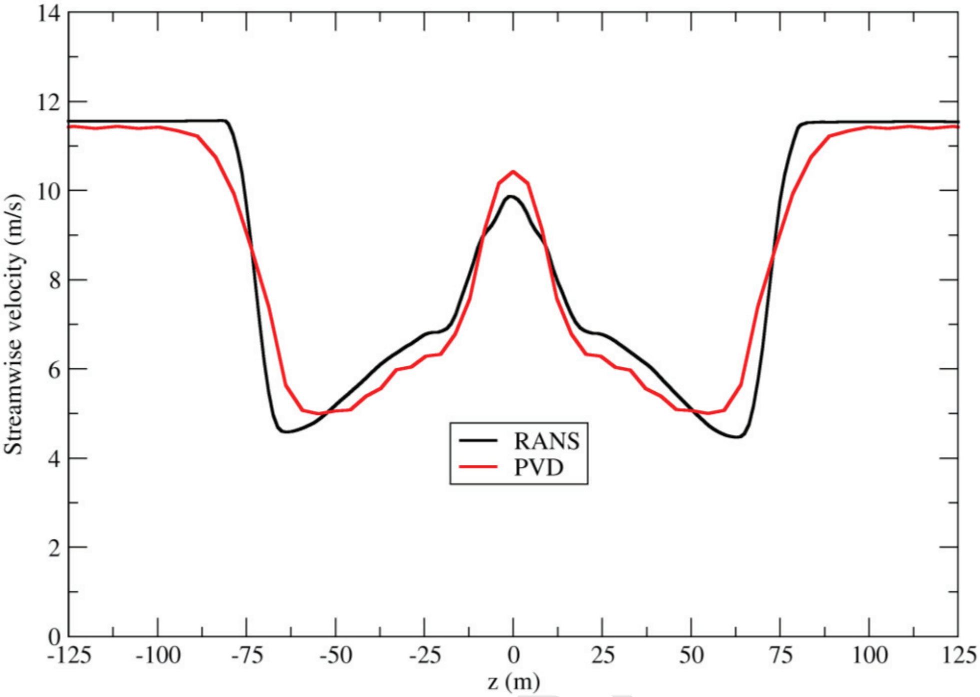


Figure 9: Streamwise velocity across the rotor plane two rotor diameters downstream of the turbine

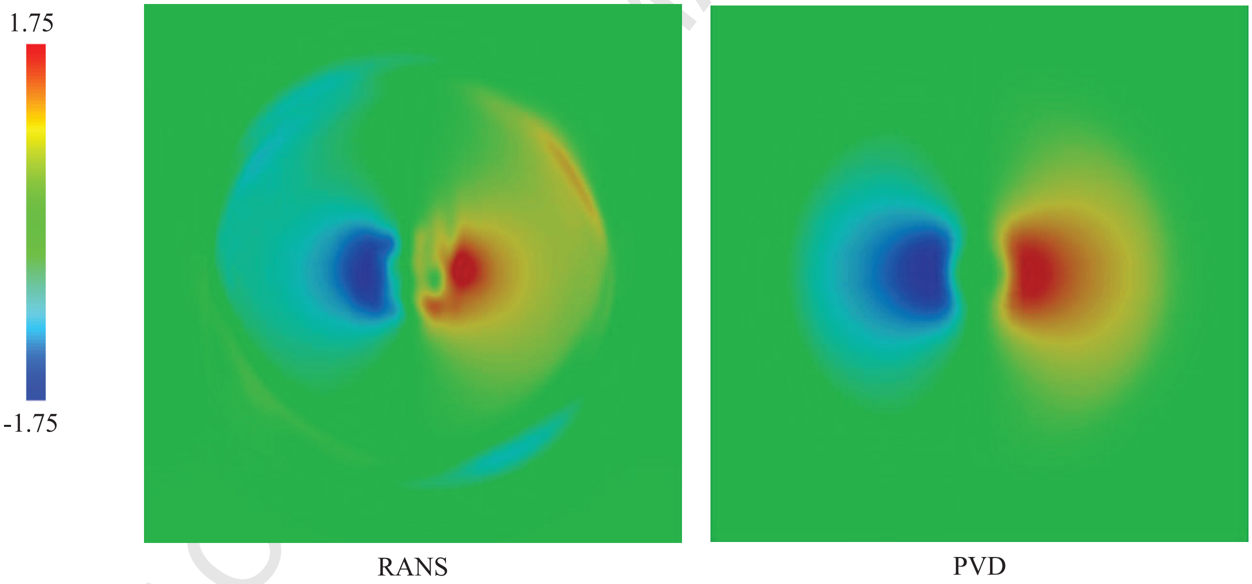


Figure 10: Contours of transverse (*y*-velocity) velocity at two rotor diameters downstream of the wind turbine

The runtime of the PVD simulation was 88 minutes on a desktop computer. The unsteady Navier-Stokes simulation required 48 cores of an in-house cluster with processors comparable to the ones on the desktop computer. This runtime was over 202 hours, bringing the computational cost to around 9,700 processor hours.

E. Wind turbine array

The PVD model was further tested in a simulation of flow through an array of turbines, including comparison with a blade-resolved RANS simulation for the same flow conditions. The test case employed three of the NREL 5-MW turbines used in the previous test case. Two turbines were positioned upstream with a lateral spacing of 1.5 rotor diameters. The third was placed 5 rotor diameters downstream and centered between the upstream turbines

(0.75 rotor diameter away from both turbines in lateral direction). The inflow speed is 11.4 m/s and all turbines were operating at constant 12.1 rpm. The domain of the simulation is 2000 m in the *y* and *z* directions. The cross-sectional PVD grid contains 22,801 nodes with spacing ranging from 48 m at the boundaries to 3.93 m near the center. The spatial marching step size was chosen as 10 m, and the PVD simulation was conducted for 200 steps.

The PVD simulation was run on a Linux desktop computer and the run time for 200 steps was slightly over 2 hours (4.56 million total grid nodes). The RANS simulation for three turbines required a very large mesh (57.2 million grid nodes), and the simulation required 60,000 total core hours using 500 cores of the in-house cluster.

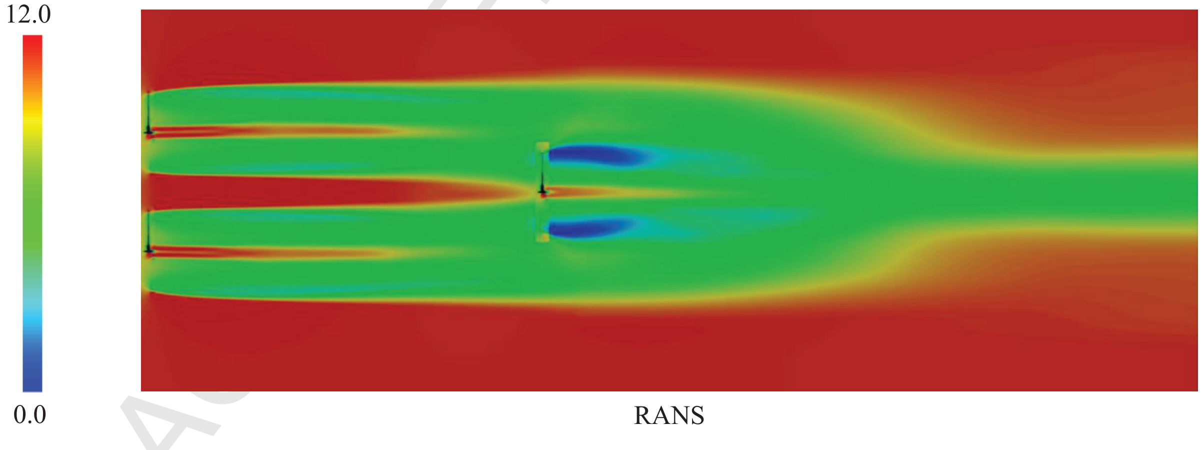
In this configuration, the flow approaching the first two turbines is uniform, but the third turbine experiences a nonuniform approaching flow caused by the wakes of the first two turbines. Table 3 shows the torque and thrust values for the third turbine computed using the PVD and RANS methods. Figure 11 shows the contours of streamwise velocity on a plane passing through the hub of all the three turbines.

It is first observed that the torque predicted by the PVD simulation is over-predicted by 347 kNm, relative to the RANS simulation. A similar over-prediction of 180 kNm was observed in the previous test case, possibly due to the coarse blade discretization of 17 points utilized in the FAST code and also used in this case. The other reason for higher torque prediction is the absence of the hub in the PVD model, which causes the velocity at the center of the wake to be larger in PVD simulations (Figure 8 and Figure 9). The higher velocities in the wake of the upstream turbines lead to higher torque production from the third turbine.

A second observation concerns the sudden RANS wake dissipation in Figure 11, beginning about five rotor diameters downstream of the third turbine. This was also seen in Figure 7 and is again attributed to inadequate local grid resolution. The very large RANS mesh required for rotating blades precluded adequate grid resolution in this downstream wake region. The PVD model does not suffer from this drawback and the wakes persist and expand far downstream.

Table 3: Thrust and torque values for the third turbine (PVD vs. RANS)

	RANS simulation	PVD simulation
Thrust (kN)	641	624
Torque (kNm)	2735	3082



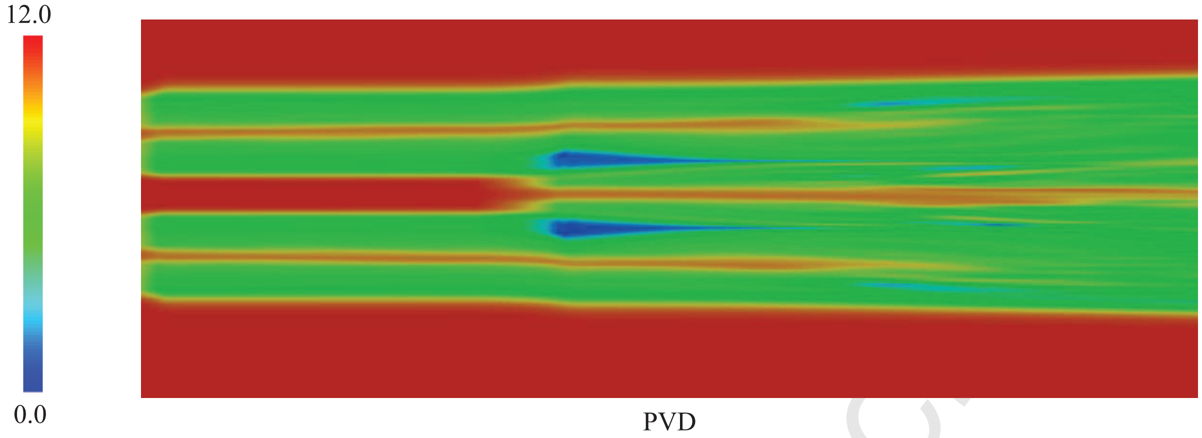


Figure 11: Contours of streamwise velocity showing the flow through the three turbine array

VI. Summary and Conclusions

It is encouraging that the wake velocity profiles predicted by the PVD method in the intermediate-wake region were in reasonable agreement with corresponding wakes obtained by time-averaging the solution from an unsteady RANS simulation with full rotor resolution. This demonstrates that the PVD method with the turbine model is reasonably effective, given differences in physical modeling fidelity, grid topography/density requirements, and computer resources

There are at least two advantages of the PVD model when compared with existing parabolic models. First, the PVD approximation is valid for flows with large streamwise vorticity and secondary velocity, and it can incorporate wake rotation caused by the turbine. Secondly, by modeling the turbine using distributed forces from a time-averaged BEM turbine model, the steady spatial marching solution can be applied to flow both upstream and downstream of the turbine. The solution can thus be initialized without prior knowledge of an upstream wake flow.

The present PVD model is several orders of magnitude faster than a RANS or LES code. As an example, for the wind turbine case presented here, the runtime for the PVD model is 0.00258 seconds per node whereas the RANS simulation takes 2.4 seconds per node. The PVD model also appears to be less dissipative for streamwise vortex and secondary flow resolution.

Given the low computational cost and observed accuracy of the solutions, the present PVD model is believed to have the potential to play an important role in wind turbine applications, provided turbulent transport is modeled adequately. PVD simulations can give a good low-cost estimates of radial and circumferential velocity distributions in the far wake velocity field. This capability is especially useful in simulations of multiple wind turbines in wind farms, where turbines operate in the wakes of upstream turbines. The cost of these PVD simulations can be orders of magnitude less than a RANS or LES simulation due to very large savings in gridding complexity, computational problem size, and computer resources, making them both feasible and attractive.

Acknowledgments

The research presented here was supported by the National Science Foundation (NSF) through a grant (CBET # 1236124) with Dr. Ram Gupta and Dr. Gregory Rorrer as technical monitors. This support is gratefully acknowledged. Images were created and data extractions carried out using FieldView as provided by Intelligent Light through its University Partners Program.

Appendix – Description of the Navier-Stokes Solver

The RANS solver used for comparison with the PVD solutions is implemented in an in-house code called Tenasi [31]. The solution algorithm is a node-centered, finite volume, implicit scheme applied to general unstructured grids with nonsimplical elements. The flow variables are stored at the vertices, and surface integrals are evaluated on the median dual surrounding each of these vertices. For the present study, governing equations for incompressible [32] flow are solved. The inviscid fluxes are evaluated using a Roe-averaged approach, while the viscous fluxes are evaluated using a directional derivative approach. Second-order accuracy is achieved using variable extrapolation, with the gradients appearing in the reconstruction evaluated using an unweighted least squares approach. The higher order reconstruction approach (5th and 7th order) based on WENO schemes is also available in the solver with some limitations [33]. Gradients appearing in the viscous fluxes are evaluated using a weighted least squares approach. A one-equation Spalart-Allmaras model was used here for consistency with the PVD simulations. The turbulence model is loosely coupled with the mean flow through the turbulent viscosity. The parallel solution procedure consists of a scalable solution algorithm implemented to run efficiently on grid subdomains distributed across multiple processes and communicating through MPI. The algorithm has multiple nested kernels viz. time step, Newton iteration, LU/SGS iteration etc., and the subdomain coupling is at the innermost level, i.e., in the solution of the linear system. A block-Jacobi type updating of the subdomain boundaries ensures efficient parallelization with a small incremental cost incurred in terms of sub-iterations required to recover the convergence rate of the sequential algorithm.

Relative motion between the wind turbine and the tower/tunnel walls is implemented using a sliding interface. The underlying concept is to extrude the pairs of sliding interfaces into adjoining domains and to interpolate for the flow variables and their gradients at the newly created nodes. The sliding interfaces are not merged in any way, and although this has the potential to lead to multi-valued solutions at a given point, this has not been observed in practice. Points that are very close to each other have approximately the same solution values. Further details on parallel implementation of the sliding interface algorithm are given by Hyams et al. [34,35].

This Navier-Stokes solver has been verified and validated for wind turbine simulations including single and multiple turbines with sizes ranging from model scale to utility scale [36–39]. The blade-resolved simulations utilize a fine mesh spacing ($y^+ < 2$) normal to the solid surfaces (blades and hub) to resolve the boundary layer. A multiple reference frame capability is used to model the rotating components of the turbine in a rotating reference frame and the stationary components in a stationary reference frame.

References

[1] S.G. Rubin, J.C. Tannehill, Parabolized/Reduced Navier-Stokes Computational Techniques, Annu. Rev. Fluid Mech. 24 (1992) 117–144. doi:10.1146/annurev.fl.24.010192.001001.

[2] R.H. Pletcher, J.C. Tannehill, D. Anderson, Computational fluid mechanics and heat transfer, Third edit, CRC Press, 2012.

[3] Z. Gao, The Basis of an Industry-standard Aerodynamic Computation and its Theoretical Basis with Inferences and Applications, Procedia Eng. 67 (2013) 230–240. doi:http://dx.doi.org/10.1016/j.proeng.2013.12.022.

[4] R. Mikkelsen, Actuator disc methods applied to wind turbines, (2003).

[5] J.N. Sørensen, W.Z. Shen, Numerical Modeling of Wind Turbine Wakes, J. Fluids Eng. 124 (2002) 393–399. http://dx.doi.org/10.1115/1.1471361.

[6] W.Z. Shen, J.H. Zhang, J.N. Sørensen, The Actuator Surface Model: A New Navier–Stokes Based Model for Rotor Computations, J. Sol. Energy Eng. 131 (2009) 11002.

[7] N. Sørensen, M. Hansen, Rotor performance predictions using a Navier-Stokes method, in: 1998 ASME Wind Energy Symp., American Institute of Aeronautics and Astronautics, 1998. doi:doi:10.2514/6.1998-25.

[8] L. Vermeer, Wind turbine wake aerodynamics, Prog. Aerosp. Sci. 39 (2003) 467–510. doi:10.1016/S0376-0421(03)00078-2.

[9] B. Sanderse, S.P. Van Der Pijl, B. Koren, Review of CFD for wind-turbine wake aerodynamics, Fluid Dyn. (2011) 1–28. doi:10.1002/we.

[10] M.O.L. Hansen, H. Aagaard Madsen, Review Paper on Wind Turbine Aerodynamics, J. Fluids Eng. 133 (2011) 114001. http://dx.doi.org/10.1115/1.4005031.

[11] A. Creech, W.-G. Früh, A.E. Maguire, Simulations of an offshore wind farm using large eddy simulation and a torque-controlled actuator disc model, arXiv Prepr. arXiv1410.3650. (2014).

[12] M.J. Churchfield, S. Lee, P.J. Moriarty, L.A. Martinez, S. Leonardi, G. Vijayakumar, et al., A Large-Eddy Simulation of Wind-Plant Aerodynamics, in: 50th AIAA Aerosp. Sci. Meet. Incl. New Horizons Forum Aerosp. Expo., Nashville, 2012.

[13] J. Sitaraman, D.J. Mavriplis, E.P. Duque, Wind Farm simulations using a Full Rotor Model for Wind Turbines, in: 32nd ASME Wind Energy Symp., American Institute of Aeronautics and Astronautics, 2014. doi:doi:10.2514/6.2014-1086.

[14] P.M. Sforza, P. Sheerin, M. Smorto, Three dimensional wakes of simulated wind turbines, AIAA J. 19 (1981) 1101–1107.

[15] P.M. Sforza, W. Stasi, M. Smorto, P. Sheerin, Wind turbine generator wakes, in: AIAA Aerosp. Sci. Meet., 1979.

[16] J.F. Ainslie, Wake Modelling and Prediction of Turbulent Properties, in: Proc. 8th BWEA Wind Energy Conf., 1986: pp. 115–119.

[17] J.F. Ainslie, Calculating the Flowfield in the Wake of Wind Turbines, J. Wind Eng. Ind. Aerodyn. 27 (1988) 213–224.

[18] A. Crespo, F. Manuel, D. Moreno, E. Fraga, J. Hernandez, Numerical Analysis of Wind Turbine Wakes, in: G. Bergeles, J. Chadjivassiliadis (Eds.), Proc. Delphi Work. Wind Energy Appl., 1985: pp. 15–25.

[19] J.G. Schepers, S.P. Van Der Pijl, Improved modelling of wake aerodynamics and assessment of new farm control strategies, J. Phys. Conf. Ser. 75 (2007) 012039. doi:10.1088/1742-6596/75/1/012039.

[20] W.R. Briley, H. McDonald, Three-dimensional viscous flows with large secondary velocity, J. Fluid Mech. 144 (1984) 47–77. http://journals.cambridge.org.proxy.lib.utk.edu:90/action/displayAbstract?fromPage=online&aid=377191.

[21] T.R. Govindan, W.R. Briley, H. McDonald, General Three-Dimensional Viscous Primary/Secondary Flow Analysis, AIAA J. 29 (1991) 361–390.

[22] J.M. Jonkman, M.L. Buhl Jr., FAST User's Guide, NREL/EL-500-29798, Golden, CO, 2005.

[23] J.P. Kreskovsky, W.R. Briley, H. McDonald, Investigation of Mixing in a Turbofan Exhaust Duct, Part I: Analysis and Computational Procedure, AIAA J. 22 (1984) 374–382. doi:10.2514/3.48457.

[24] J.M. Jonkman, S. Butterfield, W. Musial, G. Scott, Definition of a 5-MW Reference Wind Turbine for Offshore System Development, NREL/TP-500-38060, Golden, CO, 2009.

[25] D.G. Hyams, An investigation of parallel implicit solution algorithms for incompressible flows on unstructured topologies, Mississippi State University, 2000.

[26] W.K. Anderson, D.L. Bonhaus, An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids, Comput. Fluids. 23 (1994) 1–21. doi:10.1016/0045-7930(94)90023-X.

[27] M. Churchfield, S. Lee, NWTC Design Codes (SOWFA), (2012). http://wind.nrel.gov/designcodes/simulators/SOWFA/.

[28] P. Spalart, S. Allmaras, A one-equation turbulence model for aerodynamic flows, in: 30th Aerosp. Sci. Meet. Exhib., American Institute of Aeronautics and Astronautics, 1992. doi:doi:10.2514/6.1992-439.

[29] F. White, Viscous Fluid Flow, McGraw-Hill Education, 2006. http://books.google.com/books?id=QJsBngEACAAJ.

[30] H. Schlichting, J. Kestin, Boundary-Layer Theory, 7th ed., McGraw-Hill Book Company, 1979.

[31] D.G. Hyams, K. Sreenivas, R. Pankajakshan, D.S. Nichols, W.R. Briley, D.L. Whitfield, Computational simulation of model and full scale Class 8 trucks with drag reduction devices, Comput. Fluids. 41 (2011) 27–40. doi:10.1016/j.compfluid.2010.09.015.

[32] L.K. Taylor, Unsteady three-dimensional incompressible algorithm based on artificial compressibility, Mississippi State University, 1991.

[33] K. Sreenivas, A. Mittal, L.K. Taylor, L. Hereth, Higher-Order Accurate Simulations of Wind Turbine Flow Fields: A Poor Man?s Approach, in: 34th Wind Energy Symp., American Institute of Aeronautics and Astronautics, 2016. doi:doi:10.2514/6.2016-0749.

[34] D.G. Hyams, R.W. Webster, K. Sreenivas, A Generalized Interpolative Interface for Parallel Unstructured Flow Solvers, in: 40th Fluid Dyn. Conf. Exhib., Chicago, IL, 2010.

[35] D. Hyams, K. Sreenivas, R. Webster, N. Currier, A Generalized, Unstructured Interpolative Interface method for Rotor-Stator Interactions, in: 20th AIAA Comput. Fluid Dyn. Conf., American Institute of Aeronautics and Astronautics, 2011. doi:10.2514/6.2011-3700.

[36] A. Mittal, K. Sreenivas, L.K. Taylor, L. Hereth, C.B. Hilbert, Blade-Resolved Simulations of a Model Wind Turbine: Effect of Temporal Convergence, Wind Energy. (n.d.). doi:10.1002/we.1949.

[37] K. Sreenivas, A. Mittal, L. Hereth, L.K. Taylor, C.B. Hilbert, High-Fidelity Computational Simulation of the Interaction between Tandem Wind Turbines, in: 32nd AIAA Appl. Aerodyn. Conf., American Institute of Aeronautics and Astronautics, 2014. doi:doi:10.2514/6.2014-2278.

[38] A. Mittal, K. Sreenivas, L.K. Taylor, L. Hereth, Improvements to the Actuator Line Modeling for Wind Turbines, in: 33rd Wind Energy Symp., American Institute of Aeronautics and Astronautics, Orlando, 2015. doi:doi:10.2514/6.2015-0216.

[39] W.E. Hassan, K. Sreenivas, A. Mittal, L.K. Taylor, L. Hereth, Blade Resolved Simulation for a Wind Farm, in: 33rd AIAA Appl. Aerodyn. Conf., American Institute of Aeronautics and Astronautics, 2015. doi:doi:10.2514/6.2015-2269.