BLADE RESOLVED SIMULATION OF WIND FARMS

By
Walied Hassan

KidambiSreenivas
Research Professor of Computational Engineering
Committee Chair

Lafayette K. Taylor
Professor (Retired) of Computational Engineering
Committee Member

D. Stephen Nichols, III
Associate Research Professor of Aerospace Engineering (Auburn University)
Committee Member

BLADE RESOLVED SIMULATION OF WIND FARMS

By

Walied Hassan

A Thesis Submitted to the Faculty of the University of
Tennessee at Chattanooga in Partial Fulfillment
of the Requirements of the Degree of
Master of Science: Engineering

The University of Tennessee at Chattanooga
Chattanooga, Tennessee

December, 2015

ABSTRACT

This research examines the feasibility of reducing the time to solution for a high-fidelity, blade-resolved CFD simulation of a wind farm using *Tenasi*. The intent is to reduce the overall cost by initializing the flow fields of the individual turbines in the wind farm using a steady state or an unsteady simulation of a single turbine. This is achieved by replicating the mesh and the solution corresponding to a single turbine repeatedly, thus initializing the solution for the entire wind farm. The wind farm consists of the DTU 10 MW Reference Wind Turbines. Results compared to the “from scratch” solutions shows a 25 – 50% reduction in computational cost for steady simulations. For unsteady simulations, it is hypothesized that the savings could be on the order of 70%.

DEDICATION

This thesis is dedicated to my loving and supportive family.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor, Dr, KidambiSreenivas, for his support, patience, and encouragement at each and every step of this memorable journey. My thanks also go to the members of my dissertation committee, Dr. Lafayette Taylor, Dr. Stephen Nichols for sharing their knowledge and providing useful suggestions at various stages of my MSc. I would also like to thank Dr. Timothy Swafford his guidance at various stages of my graduate life at SimCenter. Finally, I would like to thank each and every member of the SimCenter for providing a friendly and healthy atmosphere for the research. The support of UT SimCenter at Chattanooga is greatly acknowledge. I was fortunate to have had the opportunity to work and study alongside the team at SimCenter. I would also like to thank my loving parents, Elhadi and Salma Hassan for all the encouragements they gave me to perceive this degree. Their patience and encouragement during all the highs and the lows of this research was essential. I would also like to thank my supporting friends for their endless support during the last 2 years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The energy consumption worldwide is increasing considerably and the investments in optimizing the process of production of energy are becoming an increasing priority. The clean sources of power generation, i.e., wind and solar energy, are being considered as the world looks towards reducing greenhouse gas emissions.

The power production in the United States comes from a variety of sources. According to the U.S. Energy Information Administration (EIA)(1), the energy consumption as for year 2013 was as follows: (coal 18%, natural gas 27%, nuclear 8%, renewable sources 8% and 36% using petroleum and other liquids). Figure 1 shows that the percentages are changing continuously as coal plants are retired in favor of cleaner alternatives, i.e., natural gas, wind, and solar.

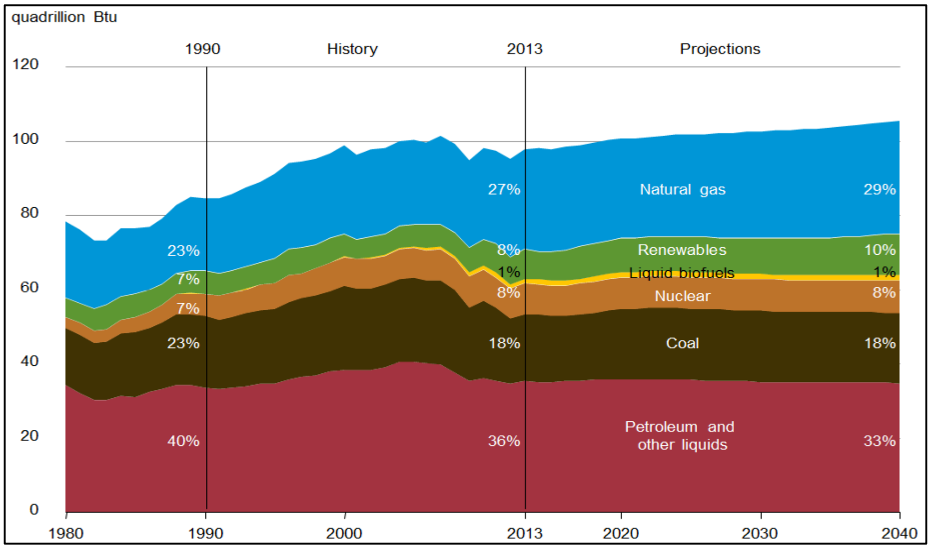


Figure 1 Energy consumption by fuel, 1980-2040

Wind power is considered one of the cleanest renewable energy mode and researchers are investing time and money to optimize its design process so as to make it cost competitive with other modes of power production.

Wind energy is harnessed using wind turbines, which convert the kinetic energy of the wind into electricity. Wind turbines have two main sections; a moving/rotating section that includes the rotor blades and the shaft, and static section which include the nacelle and the tower. Blades rotate when the wind hits the rotor and the kinetic energy is transformed into a mechanical work and then into electrical power by the generator installed inside the nacelle of the wind turbine.

Wind turbines typically do not operate in isolation. They are placed in wind farms and are usually placed with a separation distance of about 5 to 7 rotor diameters (in the axial direction) and about 2 to 3 diameters in the transverse direction. This is done so that the available land area is utilized in an optimal fashion. A downside of this proximity is that wind turbines typically operate in each others' wake. A wake is a typically a region of low kinetic energy and high turbulence levels. Consequently, the power production of the downstream wind turbine is significantly less than the upstream turbine, while at the same time, the blades of the downstream turbine experience significantly higher unsteady blade loading. During the design process of the wind farm, a layout can be created which tries to mitigate these losses and therefore increase the overall power production at a wind farm. These layouts are typically created using low fidelity methods as the optimization process requires a series of function evaluations, the function in this case being the power output of the wind farm. Using high-fidelity methods for such evaluations can be very expensive, though the power output of the final layout can be verified using a high-fidelity method.

Wind Turbine Wakes

Wind turbine flowfields have been investigated and studied both experimentally and numerically, and its wake dynamics have been a topic of intensive research for the past two decades. The challenges and motivations for those studies were due to the complicated aerodynamics before and after the turbine that comes with its stochastic and turbulent nature. Many studies demonstrated the importance of the physics of the wake region. Accordingly, the wake was classified into the near and far wake regions. The near wake region is considered to be up to one to two rotor diameters downstream of the rotor and is focused for rotor performance and power predictions. The far wake region is the region beyond the near wake region and is more focused on the effects of the wake on downstream turbines.

Many field experiments and wind tunnel measurements have been carried out with an aim towards predicting the power production and elucidating the wake physics. Both averaged data and detailed data were used to analyze the global characteristics, i.e., power and thrust(2-9) besides understanding the underlying physical interaction of the wake region with the ambient flow. These included velocity measurements, tip vortex properties and wake deficit meandering. Madsen et al.(10) showed that it is greatly acknowledged through measurements the dynamical large scale phenomenon, i.e. wake deficit meandering, may contribute to the increased loads of those turbines operating in the wake of others. Thomsen et al.(11) and Bingol et al.(12) separately interpreted the phenomenon as being attached to the traverse velocity fluctuation. Also Medici et al.(13) observed a clear large scale vortex shedding that might be connected with wake meandering. These field measurements are limited by the fact that there is no control of the ambient flow conditions. In an attempt to mitigate this, several wind tunnel studies have been carried out; however, it is impossible to place a utility-scale rotor in any existing wind tunnel facility around the world.

Another way to approach this problem is via numerical simulation. Numerical simulation serves as a valuable supplement to the expensive experiments because they can provide a thorough understanding of the wake and provide relatively accurate power predictions. Most previous research on wind turbines applied computational models using simple assumptions such as uniform flow and steady state conditions. There exists a wide spectrum of methods for performance prediction and modeling of wind turbines loading that is in use today. The methods can be classified into Actuator Disc (AD), Actuator Line (AL) or Fully Resolved (FR) methods. All methods solve the Navier-Stokes equations in the flow field. The difference arises in the way the rotor is modeled.

The idea of the AD method is to represent the rotor with a permeable thin circular disc of the same area and impose the thrust and torque of the wind turbine on the disc. These types of models have been used by several researchers to study isolated turbines(14, 15)as well as rows of wind turbines(16). Castellani et al.(17)investigated the possibility of using the AD model to simulate the wake effects, wind speed deficit and the induced turbulence in a CFD simulation of off-shore wind farms. They used a RANS formulation to calculate wind fields with a relative low usage of the computational resources. Despite the agreement of the simulated and experimental data for a single wake case, the simplicity of the model ignored the physics brought about by multiple-wake interaction beyond 3 diameters downstream of the wake region of the rotor.

Besides the AD model, the AL method was introduced with a basic principle of using the body force terms in the Navier-Stokes equations in order to distribute the forces along radiallines that represent the blades. Martinez et al.(18)used LES simulations to perform a comparison where AD and AL models were used for power prediction as well as studying the wake velocity deficit. AL model showed a non-symmetric wake with respect to center-line and also captured the vorticity structures near the rotor, at the root and the tip. On the other hand, AD model showed a symmetric wake. Moreover, AD model required a tip-loss correction to be used.One common similarity was the grid dependency, where the predicted pwer decreased as the grid resolution decreased.Churchfield et al.(19)carried out a time-accurate simulation on the operating Lillgrund wind plant with 48 multi-MW turbines. The research was based on a large-eddy simulation (LES) while using the AL method. The overall efficiency was well captured and the performanceresults agreed well with field observations downstream untilthe fifth turbine in awind plant array, however, beyond the fifth-row turbine, the wake meandering resulted in a power over-prediction errors of 25-40%. Moreover, such simulation required significant computational resources (over one million processor-hours to model 10 minutes of operation).

The AL method is a significant improvement over the AD method; however, it faces significant issues that are still being addressed. One of the main issues for both the AD and AL models is that the solution is significantly dependent on the grid resolution, i.e. the power decreased as the grid coarsened(18). Also, the forces distribution along the actuator lines was controlled by a Gaussian function that establishes a projection width. Using a uniform distribution didn’t resolve the blade well especially near the tip(18). Many improvements were introduced, e.g. Shives and Crawford suggested an improvement to the width to be a function of the local chord. The actuator line model originally models the rotor only, i.e. the blades. An issue of modeling the hub, nacelle and the tower has arise that an incomplete representation of the wake comes with ignoring the hub, nacelle and the tower. Churchfield et al.(20) found that the wake broke down quicker than the blades-only actuator models which affected the turbine operation predictions.

The fully resolved method overcomes these limitations by modeling the rotor in its entirety, i.e., no assumptions are made and the blade is not replaced by an equivalent disc or line. The only potential downside on a FR method is that the blade geometry must be available; therefore, this cannot be usually applied to commercial blades as the company might not want to share the blade geometry.

Using a FR method, Zahle(21) showed good agreement with the measurements on the NREL wind turbine coupled with overset grid method so as to allow for relative motion while keeping a reasonable grid resolution.Troldborg et al.(22) compared the result of a wake simulation of three different CFD models (AD, AL and FR) for a NREL 5MW reference wind turbine. All three models were in good agreement in predicting the axial velocity up to two diameters downstream of the turbine for a uniform inflow. Further downstream, the predictions for the full rotor model showed a faster smearing of the gradients than both the AD and AL methods.In a recent work, Sitaraman et al.(23) simulated a wind farm using a full rotor (FR) model for the wind turbine with a body-fitted mesh coupled with wake refinement to enable a high fidelity simulation. NREL 5MW turbine body-conforming grids with block-structure Cartesian grids capable of dynamic mesh adaptation were chosen for that study. The final grid had close to 300 million grid points, with the entire simulation requiring 500,000 CPU hours and utilized 3840 CPU cores.

High fidelity simulations for wind turbine have grown widely with the rapid advancement of computer speeds. Several Researchers are applying the full rotor method along with the most accurate turbulence models in order to resolve wind turbine flow fields. Although these simulations give high accuracy performance predictions, they also utilize significant computer resources and result in a high computation cost. In this research, a replication technique is developed in order to reduce the computational cost of a high-fidelity, parallel CFD modeling of a representative wind farm. The wind farm considered in this study contains different arrangementsof a DTU-10MW reference wind turbine. The replication technique represents an initial condition for a wind farm that uses agenerated solution from a single turbine case and replicated for each turbine of the farm. The replication techniqueresulted in a reduction in the computational time when compared to results from a case with free stream initial conditions.

The remainder of this thesis is organized as follows. The details of the mesh generation techniques that were developed so as to be compatible with the replication technique are described in Chapter II. Chapter III describes the anatomy of the in-house developed flow solver called *Tenasi*. The mathematical model and the numerical simulation approach employed in *Tenasi* are detailed in this chapter. Details of the replication technique used to reduce the computational cost are presented in Chapter IV. Chapter V shows the results of applying the replication technique to simulate a wind farm and a couple of test cases were used to validate the feasibility of the method in decreasing the computational costs. Tandem and triangle wind turbines arrangements (representative of a wind farm) were studied the computational savings are reported.

CHAPTER II

GRID GENERATION

The process of grid generation comprises three different techniques in order to build a body-fitted viscous layered grid on top of a surface mesh that was created inPointwise®. An off wall (near-field) anisotropic unstructured mesh is generated using the in-house developed Viscous Layer Extrusion (VLE) code. Furthermore, the farfieldmesh is generated using Voxelmesh, a non-boundary conforming, Cartesian-aligned grid generator.Due to the non-boundary conforming nature of Voxelmeshcode, the two grids (generated using VLE and Voxelmesh) are stitched together using Pointwise®. A brief description of the methodologies used by VLE and Voxelmesh are described below.

Viscous Layer Extrusion (VLE)

VLE is used to construct the necessary viscous meshes. VLE creates these meshes by extruding the triangles and quadrilaterals that comprise the surface into stacks of prisms and hexahedra, respectively. This extrusion is done on an element-by-element basis using normals and stack heights that correspond to neighboring elements. The mesh quality is maintained using a maximum bound for the element angle and a progression factor. These are user input parameters. Additionally, the user inputs a maximum bound for the expected number of layers and the code will abort the extra layers that violate the stack cut-off ratio criteria. Note that the individual stack height generated off each element is not dependent upon its neighboring elements. However, each stack is usually restricted, for quality purposes, from advancing more than two or three layers past its neighbors' stacks. That surface along with any non-viscous boundary surfaces are used to create the remainder of the mesh. The result of this process is a complete boundary layer mesh with a viscous front surface as shown in figure 2.

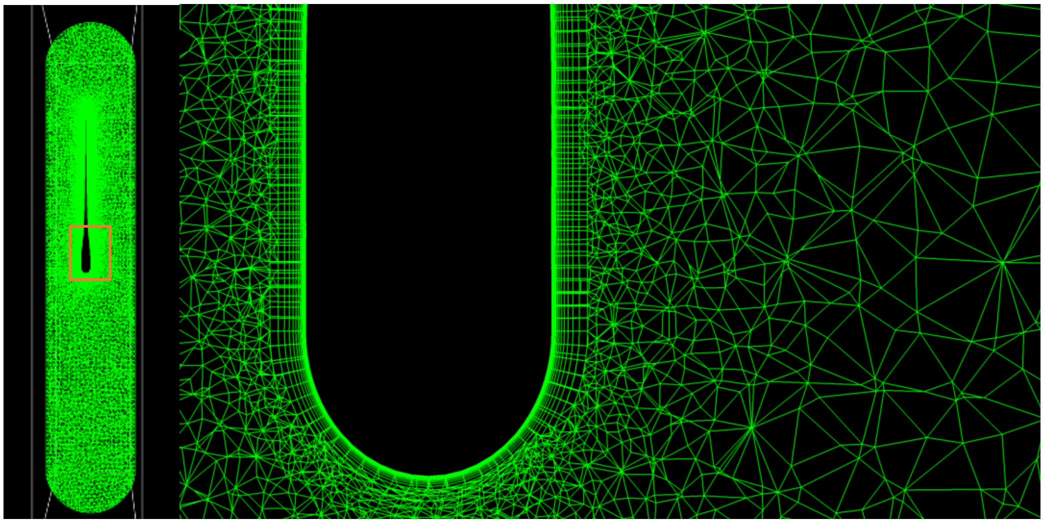


Figure 2 Viscous near-field mesh bounded with a sliding interface, created by VLE

Voxelmesh

The “inviscid” part of the mesh is constructed with Voxelmesh code. Voxelmesh is a non-boundary conforming, hierarchical, grid generation tool. It takes sizing information from the surface mesh as well as from user defined regions and, via an octree data structure, constructs Cartesian-aligned, hexahedral-dominated meshes. Figure 3 shows the block created by Voxelmesh marching off the blades sliding interface and bounded by a block representing the replication box interface.

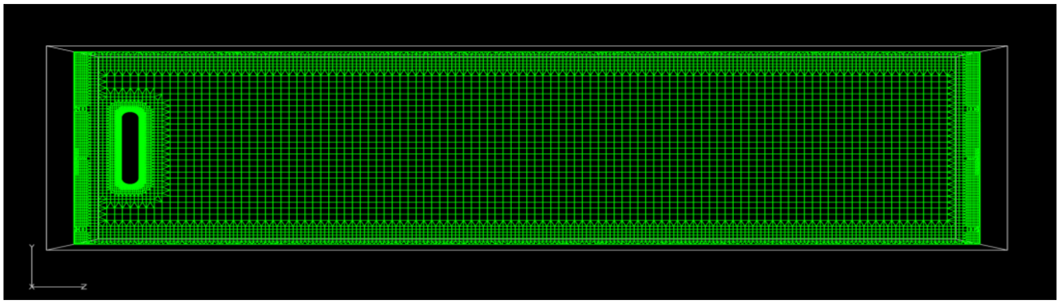


Figure 3 InviscidVoxelmesh extended from the viscous part and bounded by the replication box interface

However, since this is a non-boundary conforming process, there remains a gap between the meshes created with Voxelmesh and VLE. Pointwise is used to fill in this gap region with tetrahedra.

Sliding Interface Mesh

The simulation of a wind turbine or a wind farm (consisting of multiple wind turbines) typically involves relative motion between the turbines and the tower, turbines and the ground plane or between the turbines themselves. In the cases considered here, there was no ground plane and there were no towers; however, multiple wind turbines were simulated. In order for this to work correctly, a sliding interface mesh needs to be generated. The sliding interface mesh consists of two parts that “slide” past each other. The grids for the two parts are generated using a combination of Pointwise, VLE and Voxelmesh. Figure 2 shows the mesh inside the sliding interface along with the VLE generated viscous mesh. Figure 3 shows the mesh outside of the sliding interface, which was generated using Voxelmesh. The two parts share a common interface (the sliding interface), though the discretization on either side of the sliding interface does not have to be the same. Further details on the implementation of the sliding interface and the means of coupling the two sides of the interface are discussed in the following chapter.

CHAPTER III

MATHEMATICAL & NUMERICAL MODELS

This research work utilizes *Tenasi* to perform the simulation for the wind farm. The *Tenasi* flow solver has been developed in-house at the SimCenter: National Center for Computational Engineering and is capable of solving a wide spectrum of flow regimes. The solver is designed based on a finite volume, node-centered, implicit scheme. High-resolution fluxesare utilized to approximate the inviscid fluxes which are integrated within their control volumes built from the median dual surrounding each vertex of the mesh. Newton relaxation is used for time accurate solutions. The unstructured solver can utilize iso- / aniso-tropic three-dimensional mixed-element grids; isotropic tetrahedral elements are generated in regions far from the surfaces while prisms and hexes are extruded from the triangle or quad surface elements for high-gradient boundaries giving a rise to a highly anisotropic grid near these surfaces.

Mathematical Model

The *Tenasi* flow solver is built to solve field equations in a system of conservation laws. In this work, the incompressible form of the 3-D Navier-Stokes equations are solved. The non-dimensionalized version of the equations can be written in the integral form shown below:

$$\frac{\partial}{\partial t}\int_{\Omega} Q dV + \int_{\partial\Omega} \vec{F} \, . \, \hat{n} dA = \int_{\partial\Omega} \vec{G} \, . \, \hat{n} dA + S$$

where$Q$ is thevector of conserved variables, $\vec{F}$and$\vec{G}$ are the inviscid and viscous flux vectors respectively, and S is the source term. Specifically,

$$Q = \begin{pmatrix} P \\ u \\ v \\ w \end{pmatrix}$$

$$\vec{F} = \begin{pmatrix} \beta u \\ u(u - V_x) + P \\ v(u - V_x) \\ w(u - V_x) \end{pmatrix} \hat{\imath} + \begin{pmatrix} \beta v \\ u(v - V_y) \\ v(v - V_y) + P \\ w(v - V_y) \end{pmatrix} \hat{\jmath} + \begin{pmatrix} \beta w \\ u(w - V_z) \\ v(w - V_z) \\ w(w - V_z) + P \end{pmatrix} \hat{k}$$

$$\vec{G} = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{zx} \end{pmatrix} \hat{\imath} + \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{zy} \end{pmatrix} \hat{\jmath} + \begin{pmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \end{pmatrix} \hat{k}$$

$$\tau_{xx} = 2(1 + \mu_t)(u_x)$$

$$\tau_{yy} = 2(1 + \mu_t)(v_y)$$

$$\tau_{zz} = 2(1 + \mu_t)(w_z)$$

$$\tau_{xy} = \tau_{yx} = (1 + \mu_t)(u_y + v_x)$$

$$\tau_{xz} = \tau_{zx} = (1 + \mu_t)(u_z + w_x)$$

$$\tau_{yz} = \tau_{zy} = (1 + \mu_t)(v_z + w_y)$$

The artificial compressibility parameter (β) is typically set to 5. The equations above have been non-dimensionalized with respect to the free stream velocity (U_∞), characteristic length (L) and the free stream kinematic viscosity($\nu_\infty = \mu_\infty / \rho_\infty$). Thus, $Re = U_\infty L / \nu_\infty$, and the non-dimensionalized pressure $P = (P^* - P_\infty)/(\rho_\infty U_\infty^2)$ where P^* is the local static pressure.

Numerical Model

Tenasi has many built-in libraries that allow subject matter experts to simulate numerous application domains. The flow solver has been primarily built to model fluid dynamics problems with arbitrary Mach number. It simulates high-Reynolds number viscous flows by taking advantage of the turbulence modeling capabilities (various eddy viscocity models, Reynolds Stress models and LES). Moreover, a generalized interpolative interface technique has been added to allow simulation of components in relative motion, such as moving control surfaces and rotating parts.*Tenasi* has been used in a wide range of real world applications such as full-configuration aircraft, truck drag reduction and staged turbomachinery.A brief description of the various components of *Tenasi* are included below.

Residual computation

The flux integral is evaluated around the control volume (median dual; Figure 4) formed by the interior and boundary edges. High resolution fluxes based on the Roe averaging technique are utilized to approximate the inviscid fluxes, i.e.,

$$\oint_{\partial\Omega_i} \vec{F}\,.\,\hat{n}dS \;\approx\; \sum_{j\in N(i)} H_{ij}\,dS_{ij}$$

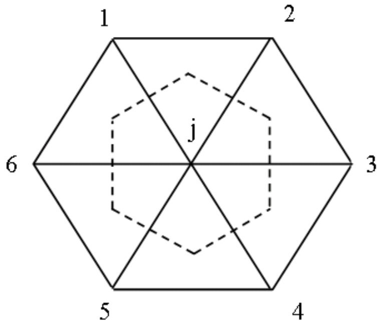


Figure 4 Two-dimensional control volume with flux faces linked from the median duals

whereH_{ij} is the numerical flux through a face connecting nodes *i* and *j*. In the present case,

$$H_{ij} = \frac{1}{2}\Big(F(Q_i) + F\big(Q_j\big)\Big) - \frac{1}{2}\|\vec{n}_{ij}\|\tilde{A}\big(Q_i, Q_j\big)\big(Q_j - Q_i\big)$$

and$\|\vec{n}_{ij}\|$ is the area associated with edge *ij*. $\hat{A}$is the flux Jacobian matrix evaluated using the Roe-averaged variables. For incompressible flows, the Roe variables can be defined using just an arithmetic average. Following Taylor [29], the three-dimensional eigensystem for incompressible flows can be written as follows:

$$\tilde{A} = \tilde{T}\tilde{\Lambda}\tilde{T}^{-1}Q$$

Defining

$$a_t = -\big[\hat{n}_x V_x + \hat{n}_y V_y + \hat{n}_z V_z\big]$$

$$\theta = \hat{n}_x u + \hat{n}_y v + \hat{n}_z w + a_t$$

$$c = \sqrt{\left(\theta - \frac{a_t}{2}\right)^2 + \beta}$$

$$c^+ = \frac{a_t}{2} + 2 \qquad\qquad c^- = \frac{a_t}{2} - 2$$

$$\phi = 4c\left[1 + \theta\left(\frac{\theta - a_t}{\beta}\right)\right]$$

$$\phi_1 = \hat{n}_x + \frac{u\theta}{\beta}\phi_2 = \hat{n}_y + \frac{v\theta}{\beta}\phi_3 = \hat{n}_z + \frac{w\theta}{\beta}$$

$$\phi_4 = v_{2y}\phi_3 - v_{2z}\phi_2\phi_5 = v_{2x}\phi_3 - v_{2z}\phi_1\phi_6 = v_{2x}\phi_2 - v_{2y}\phi_1$$

$$\phi_7 = v_{1y}\phi_3 - v_{1z}\phi_2\phi_8 = v_{1x}\phi_3 - v_{1z}\phi_1\phi_9 = v_{1x}\phi_2 - v_{1y}\phi_1$$

withV_x, V_y, and V_z being the grid speeds in the x, y, and z directions respectively. The eigenvaluesappearing in Λare:

$$\lambda_{1,2} = \theta \qquad \lambda_3 = \theta + c - \frac{a_t}{2}\lambda_4 = \theta - c - \frac{a_t}{2}$$

The right and left eigenvectors are defined as follows:

$$T = \begin{pmatrix} 0 & 0 & -c^{-} & -c^{+} \\ 2v_{1x} & 2v_{2x} & \hat{n}_x + u\lambda_3/\beta & \hat{n}_x + u\lambda_4/\beta \\ 2v_{1y} & 2v_{2y} & \hat{n}_y + v\lambda_3/\beta & \hat{n}_y + v\lambda_4/\beta \\ 2v_{1z} & 2v_{2z} & \hat{n}_z + uw/\beta & \hat{n}_z + w\lambda_4/\beta \end{pmatrix}$$

$$T^{-1} = \frac{2}{\phi}\begin{pmatrix} (c/\beta)[-u\phi_4 + v\phi_5 - w\phi_6] & c\phi_4 & c\phi_5 & c\phi_6 \\ (c/\beta)[u\phi_7 - v\phi_8 + w\phi_9] & -c\phi_7 & -c\phi_8 & -c\phi_9 \\ (\phi/4c) - (c^{+}/\beta)(\theta - a_t) & c^{+}\hat{n}_x & c^{+}\hat{n}_y & c^{+}\hat{n}_z \\ (-\,\phi/4c) + (c^{-}/\beta)(\theta - a_t) & -c^{-}\hat{n}_x & -c^{-}\hat{n}_y & -c^{-}\hat{n}_z \end{pmatrix}$$

Higher order spatial reconstruction

The introduction of gradients of the solution at the vertices is utilized for both the higher order reconstruction and the discretization of the viscous fluxes if the directional derivative method is performed to compute the viscous flux in an unstructured grid. The assumption of the first order method in a typical upwind flux is to evaluate the values for the interface using the information from the left and right vertices, i.e. $Q_L = Q_i \; and \; Q_R = Q_j$. A better approximation of the solution is via a linear-reconstruction, i.e.,

$$Q_L = Q_i + \nabla Q_i \cdot (\vec{x}_e - \vec{x}_i)$$

$$Q_R = Q_j + \nabla Q_j \cdot \left(\vec{x}_e - \vec{x}_j\right)$$

where$\vec{x}_e$ is the edge midpoint position. The gradients appearing in the reconstruction are evaluated using an unweighted least-squares approach.

Noting that the point values and gradients at the two nodes making up the edge are available, it is possible to utilize a quadratic reconstruction, i.e.,

$$Q_L = Q_i + \frac{1}{4}\left(Q_j - Q_i\right) + \frac{1}{2}\nabla Q_i \cdot (\vec{x}_e - \vec{x}_i)$$

$$Q_R = Q_j + \frac{1}{4}\left(Q_i - Q_j\right) + \frac{1}{2}\nabla Q_j \cdot \left(\vec{x}_e - \vec{x}_j\right)$$

For grids or regions of grids that are Cartesian like, i.e., are made up of fairly straight lines, a higher order reconstruction based on a WENO approach can be used. In this instance, the grid is processed to identify edges that can support (meaning that a particular edge has enough points on either side of edge that are lined up with the edge within a certain tolerance) either a 5th order or a 7th order WENO stencil. Any edges that are not capable of supporting the higher order stencils are defaulted to the quadratic reconstruction. The weights appearing in the 5th order WENO reconstruction are calculated using the approach of Henrick et al.(24), while the 7th order WENO weights are calculated using the method proposed by Titarev and Toro(25).

It should be noted that the current implementation does not attempt to traverse subdomain boundaries when building the WENO stencils. The reason for this is that the amount of data transfer across the subdomain boundaries would become prohibitive if multiple layers of data needs to be transferred. This is not an issue with the linear and quadratic reconstructions as the point and gradient data are exchanged at subdomain boundaries. The impact of the block boundaries on reconstruction order is shown in Figure 5, with red representing 7th order reconstruction and blue representing quadratic reconstruction.

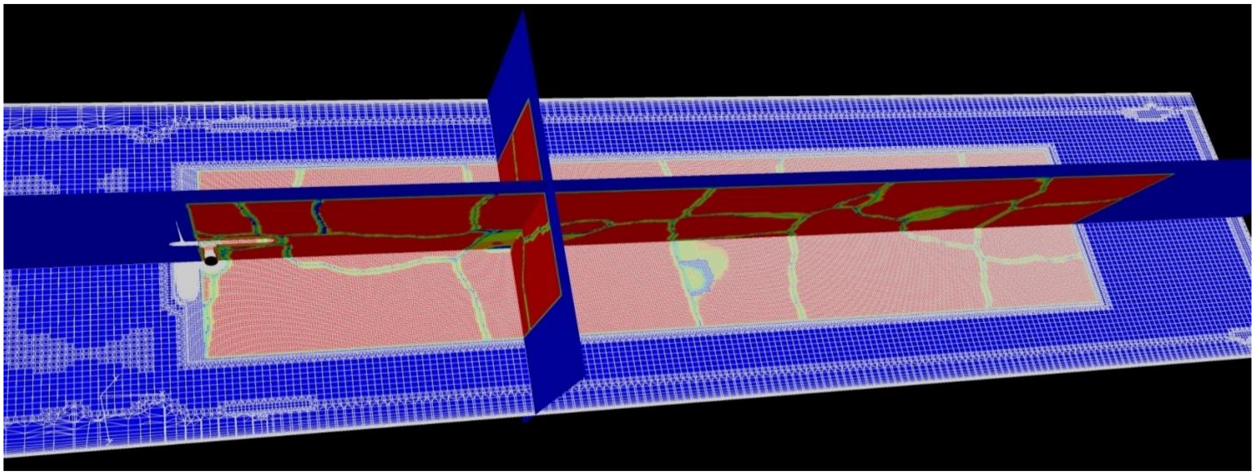


Figure 5 Impact of subdomain boundaries on reconstruction order

Viscous fluxes and turbulence modeling

The gradients appearing in the viscous fluxes are calculated using the grid-transparent, directional derivative based method. Furthermore, turbulence is modeled in a loosely coupled manner by different models: Spalart-Allmaras model, the Menter SAS model, the *k-ε/k-ω* hybrid model (with and without SST), the standard *k-ε* model, the Wilcox *k-ω* model, and the Wilcox Reynolds Stress model. In addition, DES modes are available for the Spalart- Allmaras, the Menter SAS and the *k-ε/k-ω* hybrid models[9].

Time evolution

Once the fluxes have been calculated, the solution needs to be advanced in time so that time marching can take place. Time marching can be used for steady as well as unsteady problems. In both cases, it is beneficial to introduce a pseudo-time derivative and drive the resulting system to convergence, i.e., the original incompressible Navier-Stokes equations are modified as follows:

$$\frac{\partial Q_i}{\partial t} + \frac{\partial Q_i}{\partial \tau} + R(Q_i, Q_j) = 0$$

where*R*represents the spatial discretization residual, *t*represents the physical time, and τrepresents the pseudo-time. This is also referred to as a Dual Time-Stepping method.

Steady state solution

For a steady solution, the physical time derivative is zero (by definition). Thus the above equation becomes

$$\frac{\partial Q_i}{\partial \tau} + R(Q_i, Qq_j) = 0$$

This equation is solved using an implicit formulation, i.e.,

$$\frac{\Delta Q}{\Delta\tau} + R^{n+1}\left(Q_i, Q_j\right) = 0$$

The solution of such equations requires the evaluation of the residual at the new time level, i.e., R^{n+1}. A problem arises when we try to directly solve the equation without knowing the value of Q^{n+1}. Hence, we can linearize the residual at the new time step R^{n+1} in the equation about the current time level, i.e.

$$R^{n+1} \approx R^n + \left(\frac{\partial R^n}{\partial Q^n}\right)\Delta Q^n$$

where$\Delta Q^n = Q^{n+1} - Q^n$ and the term $\left(\frac{\partial R^n}{\partial Q^n}\right)$ represent the flux Jacobian.

Since the objective is to get to the steady state solution as quickly as possible, the pseudo-time step for each control volume is allowed to vary. This is called a constant CFL approach or also referred to as local time stepping. The pseudo-time step $\Delta\tau$for each control volume$i$, is defined as:$\Delta\tau_i = CFL\frac{V_i}{\sum_{j=1}^{N} max(|\lambda_{ij}|)\cdot\|\vec{n}_{ij}\|}$

where CFL is the Courant-Friedrich-Lewy number, V_i is the volume of the control volume, λ_{ij} and$\vec{n}_{ij}$ are the eigenvalues and area associated with edge *ij* and *N* is the number of nodes connected to node *i*.

Unsteady time integration

For unsteady time integration,a backward differentiation formula (BDF2) discretization of the time derivative is considered. This coupled with a Newton sub-iteration approach(26), is used to ensure time accuracy. The stability and convergence of the resulting scheme is bolstered by adding a pseudo-time derivative (similar to the dual time stepping approach) and iterating it to convergence.

Boundary Conditions

Characteristic variable based boundary conditions are used for the farfield boundaries, while no-slip conditions are enforced for the viscous surfaces.

Relative motion

In the process of modeling bodies or mesh blocks in relative motion, there are several techniques that have been introduced. Overset grid or Chimera and remeshing approaches are of the most popular among them, however, they come with a large overhead due to the process of cutting and interpolation, and such approaches still suffer from scalability problems; typically, the package performing the cutting is not parallel at all, which introduces a serious limitation in the problem sizes that can be reasonably handled via this approach.

The interpolative interface method centers around an extruded interface that does not require matched or similar unstructured grids on the corresponding surfaces. The underlying geometry that the distinct surface grids are discretizing at the joining interface should, however, match as well as is reasonable.

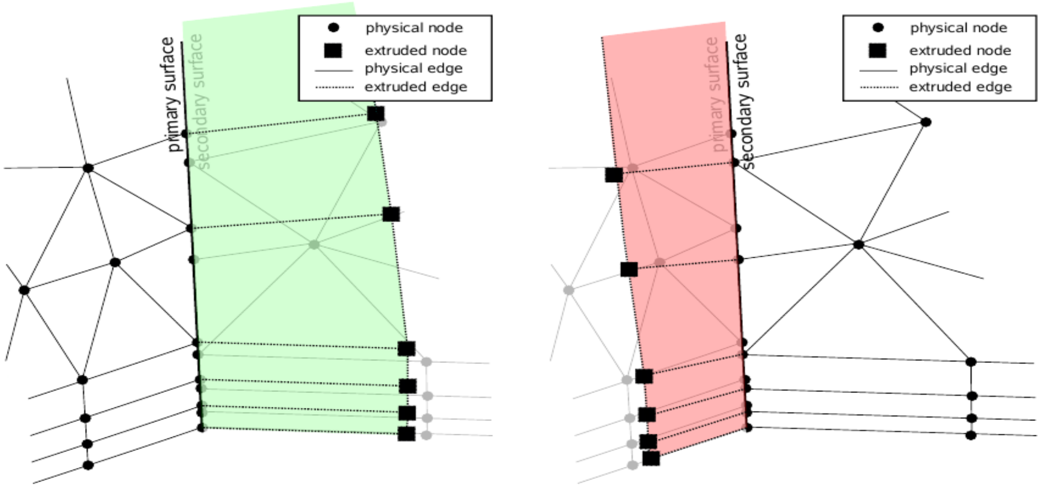


Figure 6 Extrusion process including extrusion vectors length and direction

For any two parts undergoing a relative motion, this generalized interface is utilized between the two blocks (can be referred to as rotor and stator), so that each can be solved within its own frame of reference. This interface supports the sliding of the rotor relative to the stator, so that solution variables are transferred from one domain to the other accordingly. Bridging the solution between the rotor and stator sections is achieved via interpolations from mesh extrusions constructed from the interpolative interface in both sides (see Figure 6).The extrusion step creates the nodal and element connectivitiesthat are used to couple two parts of a mesh. The interpolative interface needs to be rebuilt every time the grid is modified, i.e. relative motion, load balancing or beginning of a solution run. Note that the interpolative interface can also be used in cases where there is no relative motion between two grid blocks, but they share an interface which can be matching or non-matching).

The newly created nodes need to be assigned values (or their values updated) every time a new nodal quantity is computed within the solver. For example, when the gradients are computed in preparation for the residual calculation, the gradients at the extruded nodes must be interpolated for. Also, when the solution Q is updated with ΔQ, again the latest Q is interpolated for on the extruded nodes. These interpolations perform the task of “stitching” the two sides of the interface.

CHAPTER IV

REPLICATION TECHNIQUE

A wind farm typically consists of many turbines. A blade resolved simulation of a wind farm requires the use a mesh that contains hundreds of millions of grid points and very large supercomputers. Even with this level of hardware resources, a single simulation (with a specified wind direction) could take several days to weeks. This is usually not practical; however, under certain conditions, such a simulation would provide a wealth of data that can be used for improving the performance of the wind farm. The aim of this research is to explore ways to reduce the amount of time required to carry out a blade resolved simulation of a wind farm.

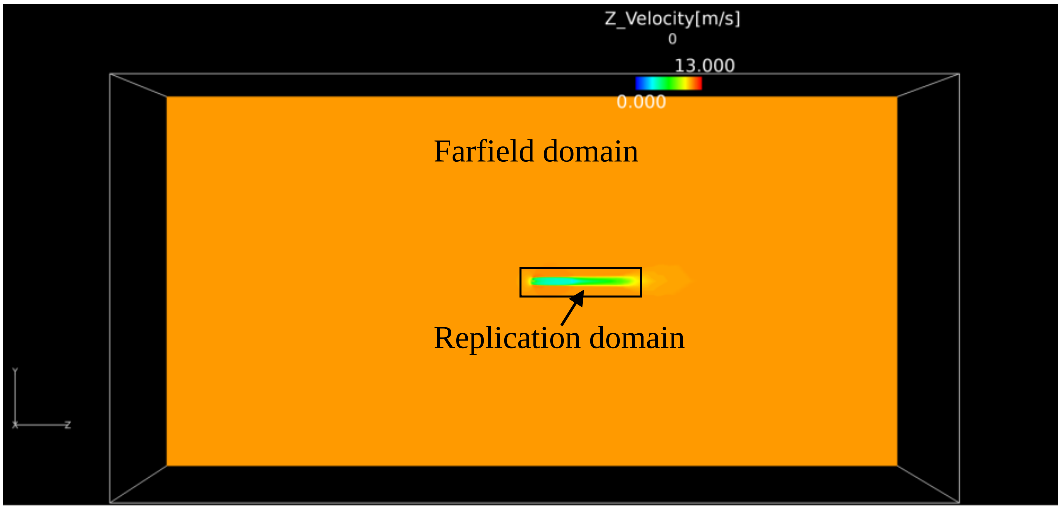
For a blade resolved wind farm simulation to be practical, a computational cost reduction technique (called the replication technique) has been developed and the details are provided here.

Replication Technique

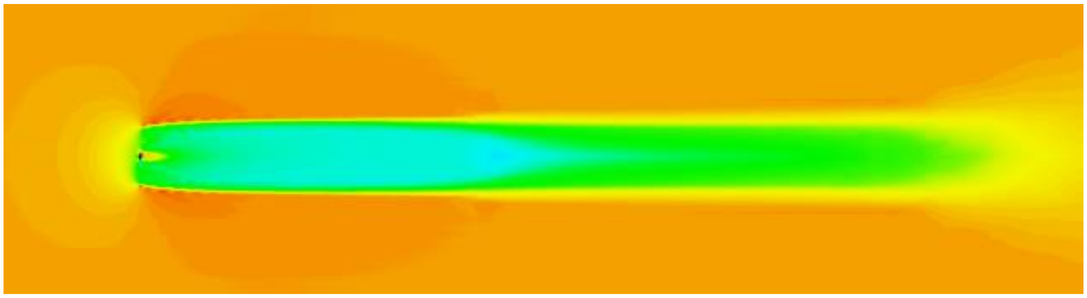
In a typical simulation, the flowfield is initialized to ones and zeros and the simulation is carried out until the power output of the various turbines reach a periodic value. In a wind farm, the turbines are operating in each others wakes. Capturing this interaction is expensive as the wake generated from the first row of turbines has to travel to the second row of turbines before it can interact with them and affect their power production. Similar arguments can be made for the third row of turbines and so on. A typical simulation initializes the flowfield to ones and zeros.

This means that a significant amount of simulation time is spent in convecting the wakes from the first row of turbines to the second before the second row will react to the incoming wake. Similarly, this combined wake has to travel downstream to the third row and that will take even longer in terms of computer time. These factors play a significant role in the increased cost of a wind farm simulation.

One potential way to reduce this cost is to find an approach where the wake does not take as long to convect from the first row of turbines to the second and so on. The replication technique is such an approach. The replication technique, as the name implies, is a procedure where a solution for a single turbine is replicated as many times as the number of turbines and the resulting flowfield is iterated to convergence.



(a)



(b)

Figure 7 Layouts of (a) farfield and (b) replication domains

In order to utilize the replication technique, care should be taken in building the mesh for the single turbine case. The mesh for the single turbine case is built in two parts: the first part is the farfield domain (outer box in Figure 7(a)) and the second part is the replication domain (inner box in Figure 7(b)). The farfield domain contains the farfield boundaries and the replication interface, while the replication domain includes the turbine (and associated sliding interfaces) and the replication interface. The two grids share the replication interface which is treated as a sliding interface (static) during the solution process. The axial and lateral extents of the replication domain are based on the expected axial and lateral spacing between the turbines. Any grid refinements required to preserve the wakes are included within the replication domain.

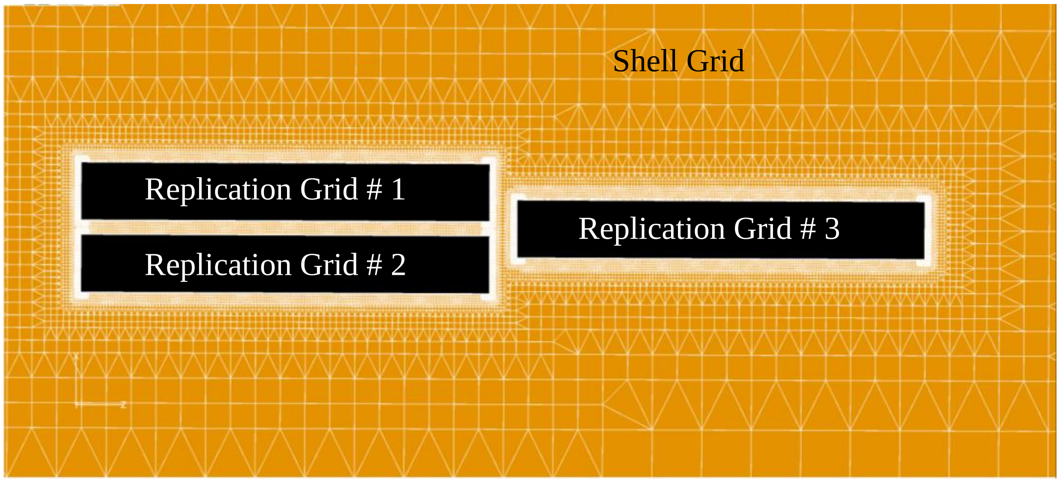


Figure 8 Shell grid for wind farm simulation (triangle configuration)

The grid for a wind farm case is created with “holes” representing the replication boxes (Figure 8). This is referred to as the shell grid. Once the appropriate grids have been built, the solution for the single turbine case can be obtained and replicated for the multi-turbine case. The procedure for this replication is given below:

(1) A converged solution is obtained on the single turbine mesh for a free stream velocity corresponding to that expected for the multiple turbine case. This solution can be a steady solution or an unsteady solution depending on the case being considered.

(2)A shell grid is generated for the multiple turbine case, but with holes representing the turbines and their wakes. This grid is initialized with uniform flow corresponding to the expected free stream velocity.

(3)The grid and solution of the replication region is extracted and duplicated (with the required coordinate offsets) into the initial solution from step (2). In the current implementation, the communication between the duplicated grids and the background grid is by the means of an interpolative interface; this can be changed such that the duplicated grids are “glued” into the background grid, thus avoiding the necessity of additional interpolative interfaces.

(4)This initialized solution is then iterated to convergence (either steady or unsteady).

CHAPTER V

RESULTS

One of the aims for this research is to study the wake physics and velocity deficit caused by the first row of turbines in a wind farm and their interactions with the downstream ones. This is accomplished by considering a couple of representative wind farm arrangements. The newly developed replication technique was used to improve the utilization of the computational resources by cutting down the computational cost. In the process of achieving this goal, the research work started with an initial step of developing an optimalmesh resolution for a single turbine.

Single-turbine mesh optimization

For this optimization, three different meshes were tested and the details are shown below. In the figures included in this section, only the region of the mesh near the rotor is shown. For all the cases, the overall mesh had extents of 10 km (transverse) x 10 km (transverse) x20 km (streamwise).

Coarse single-turbine mesh

This mesh was generated using VLE for the viscous boundary layer and Voxelmesh to mesh the farfield (Figure 9). As can be seen from the figure, this mesh did not include any wake region refinement and the grid gets very coarse relatively quickly.

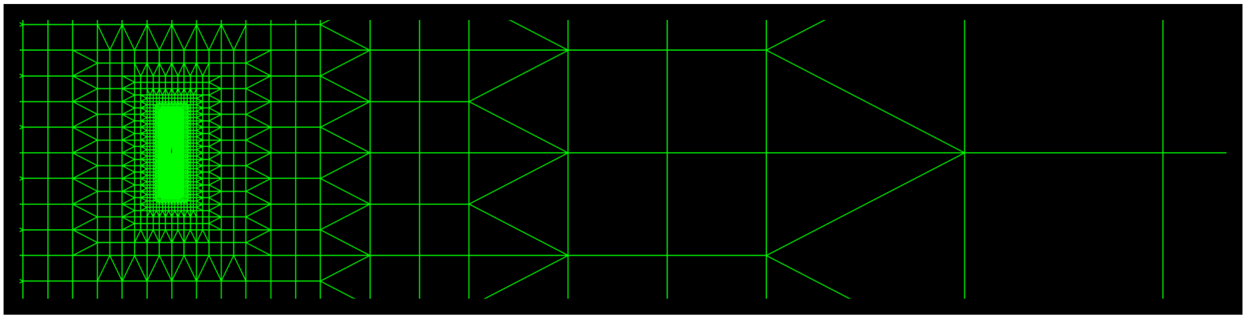


Figure 9 Coarse single-turbine mesh

Single-turbine mesh with wake region refinement

This mesh was generated using the same approach as the coarse single-turbine mesh. The main difference was the inclusion of wake refinement by means of a structured wake box (SWB; see Figure 10). The SWB extended 13diameters downstream of the turbine. This number was chosen based on the expected streamwise spacing between the turbines.

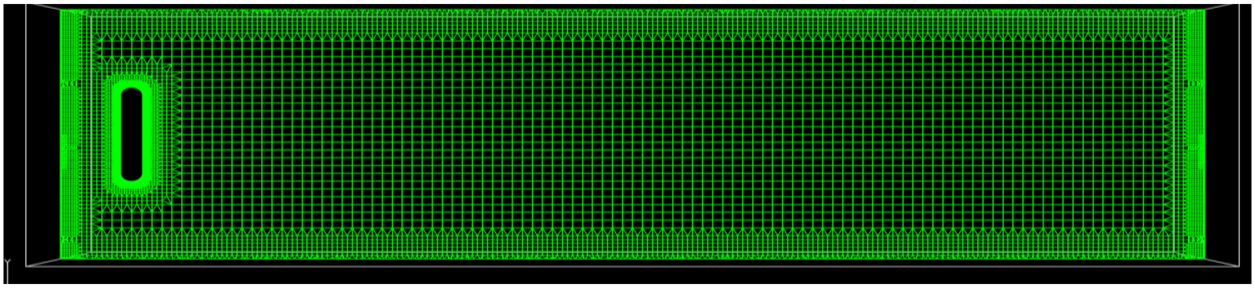


Figure 10 Structured wake box (SWB)

Single-turbine mesh with tip-vortex refinement

This mesh was built using techniques similar to the coarse single-turbine mesh. It differed from the wake-refinement mesh by including refinement for the tip-vortex region as opposed to uniform refinement downstream of the turbine. This resulted in a coarser mesh in certain regions compared to the SWB, but closer to the blades, there is a region that is significantly more refined compared to the SWB mesh (Figure 11 vs. Figure 10). The refinement for the tip vortex region extended 5.5 diameters downstream of the turbine. The replication box in this case extended 13 diameters downstream of the rotor.

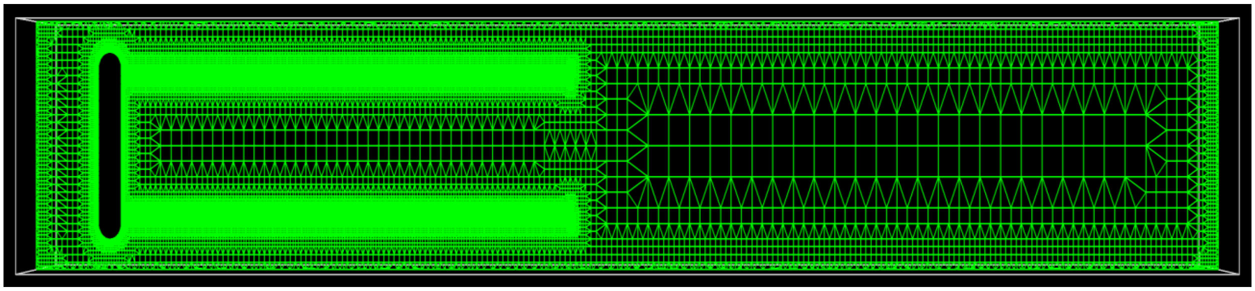


Figure 11 Wake region with tip-vortices refinement

Since the objective of these single turbine simulations was to generate solutions compatible with the replication strategy, all meshes except the coarse one included the replication box. Also, the rotor meshes (surface as well as the volume mesh within the sliding interface) were identical between the three meshes. The node counts and details of the replication boxes (if present) are shown in Table 1.

Table 1 Grid count for different single-turbine meshes

Mesh description	Total grid count (nodes)	Replication box count (nodes)
Coarse mesh	8.9 M	n/a
Structured wake box (SWB)	12.4 M	4.0 M
Tip-vortex refinement	14.7 M	6.7 M

All three meshes were simulated using the in-house flow solver, *Tenasi*,and the results are presented here. Simulations were run on 64 cores of an in-house cluster. All simulations were carried out by local time stepping (constant CFL). Simulation parametersare shown in Table 2.

Table 2 Simulation run parameters

Reference velocity (tip speed)	90 m/s
Reference length	178.3 meters
Rotor RPM	8.83
Freestream velocity	11 m/s
Number of time steps	5,000
CFL	Ramped from 1 to 50 in 100 steps
Turbulence model	Two equation k-ε/k-ω SST
Order of accuracy in space	7th order or lower depending on mesh

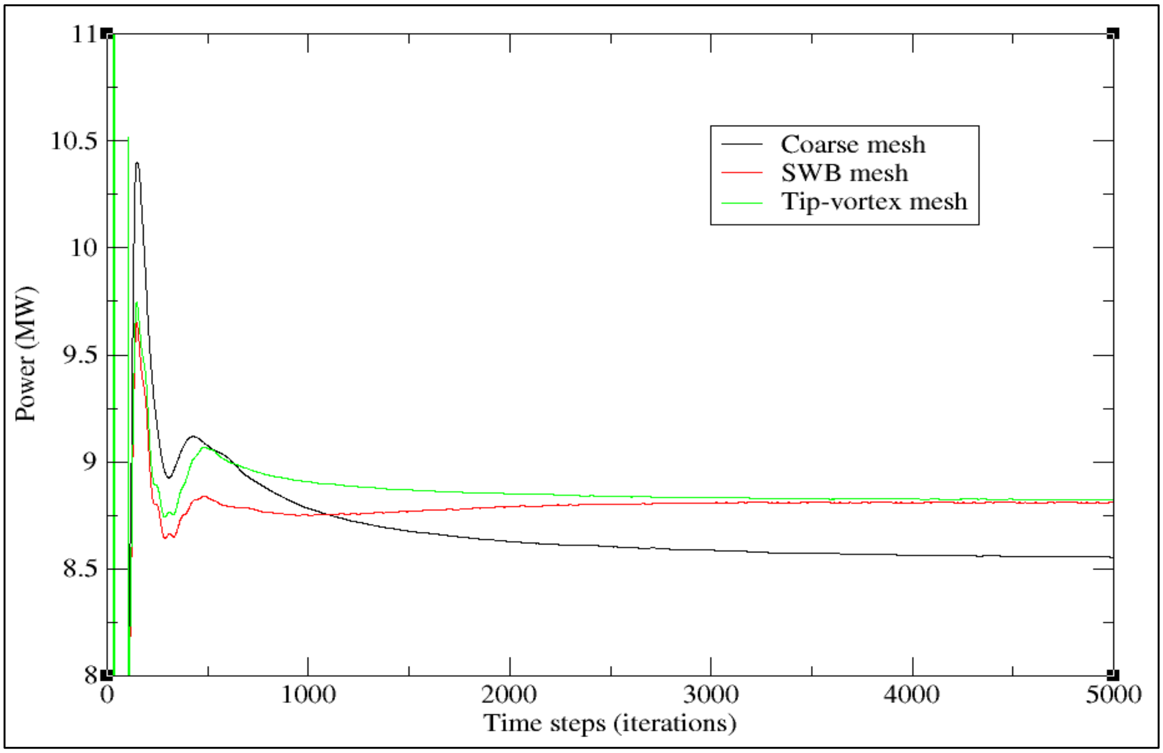
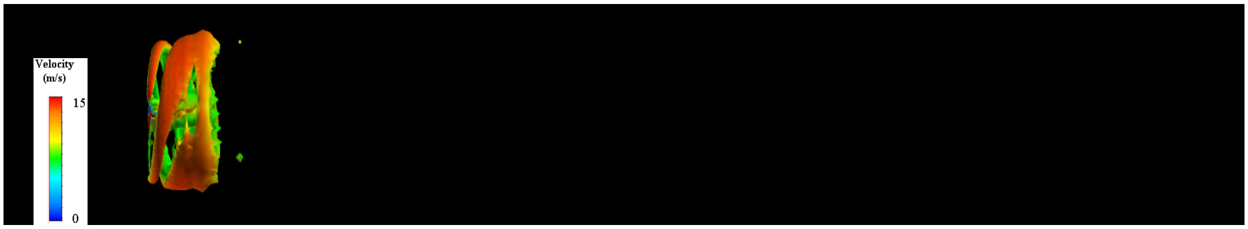


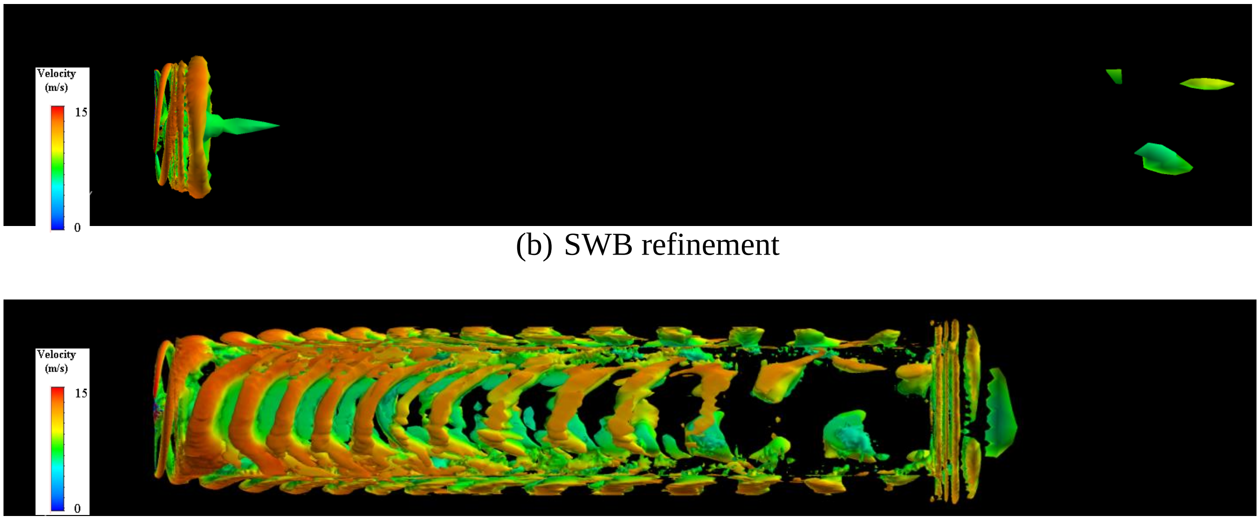
Figure 12 Steady power predictions for three run cases with different meshes

The power predictions for the three cases are shown in Figure 12. As can be seen from the figure, the presence of wake refinement (either the SWB or tip-vortex refinement) has a significant effect on the power produced. The power produced on the coarse mesh was 8.5 MW, while for the wake refinement cases, it converged to a value slightly above 8.75 MW for the tip-vortex and the SWB cases. Given that the rotor grids (surface as well as volume) are identical across all three meshes, this improvement in power production can be attributed to a better resolution of the wakein general and the tip vortex structure in particular. The resulting upwash at the blades resulted in an improvement in the power production.

Figures 13shows the iso-surfaces of the Q-criterion (iso-value = 0.001) which helps visualize the tip-vortex structure obtained for the three meshes. As can be seen from these figures, there is hardly any recognizable tip-vortex structure for the coarse mesh, with alittle more definition for the SWB case.There is a clearly defined structure for the tip-vortex refinement case. Some artifacts of the steady solution are visible in the solution. These features include a pile-up of the vorticity near the end of the grid refinement. It has been observed in other simulations carried out by the authors that these artifacts tend to get washed out once an unsteady solution is obtained. Even accounting for these artifacts, the tip vortex structure is clearly captured by the simulation.



(a) Coarse mesh



(b) SWB refinement

(c) Tip-vortex refinement

Figure 13 Iso-surface of the Q-criterion (iso-value = 0.001) colored by axial velocity

Replication (steady solutions)

Two cases, representative of wind farms, were considered in assessing the efficiency improvement obtained using the replication technique. The first case is of two turbines operating in tandem, while the second case consists of the turbines operating in a triangle configuration. As was mentioned in Chapter IV, the replication technique can utilize either a steady or an unsteady solution. In all the cases considered in this section, the solutions were steady.

Two turbines operating in tandem

Using typical wind farms layouts for guidance, a streamwise (axial) spacing of 7 diameterswas chosen. This case was run with the solution initialized to ones and zeroes(referred to as “from scratch” solution). The flow parameters used for this case were the same as those for the single turbine case, with the second turbine also rotating at the same RPM. The resulting power production is shown in Figure 14. As can be seen from the figure, the rear turbine is essentially operating as a propeller (power production = - 1 MW), i.e., it is consuming power instead of generating it while the front turbine is producing almost 8.5 MW of power. This is clearly not the intended mode of operation for the rear turbine.Given the underwhelming performance of the rear turbine, this case was not attempted using the replication procedure.

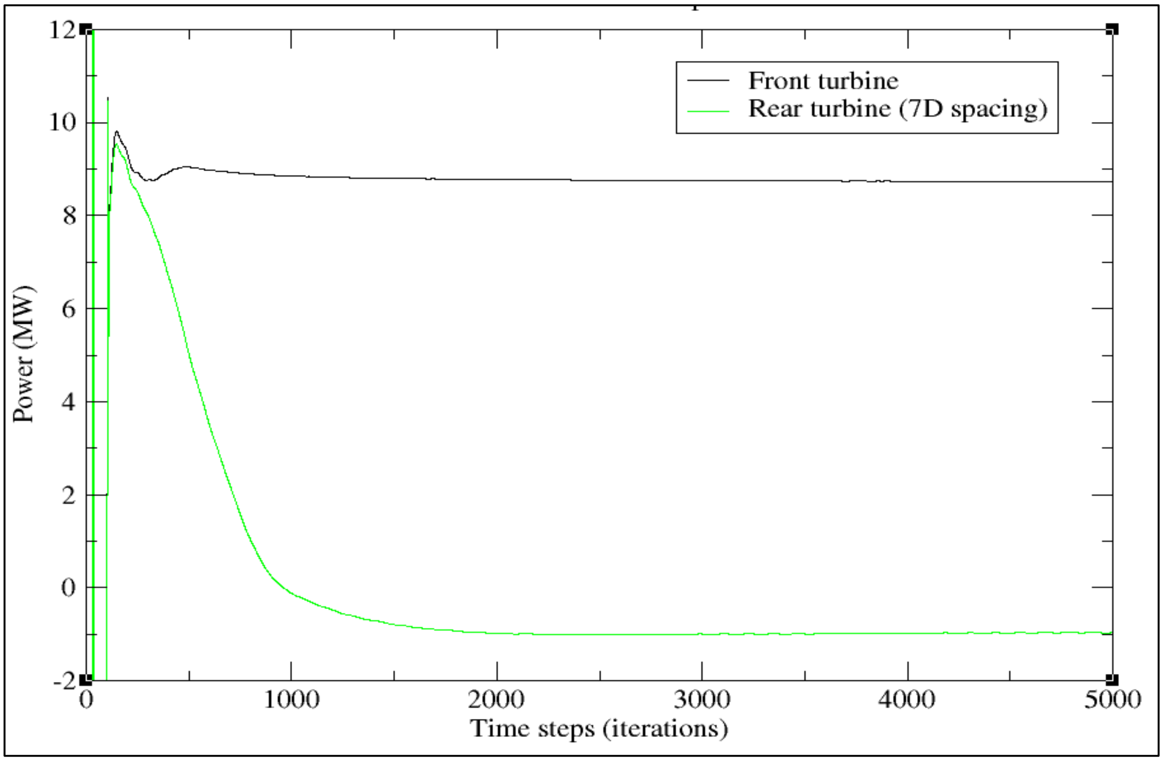


Figure 14 Steady power predictions for the tandem cases (streamwise distances = 7 diameters)

In an attempt to understand the significant underperformance of the rear turbine, the velocity field seen by the rear turbine was considered (Figure 15). Figure 15 shows an axial cut (colored by axial velocity and clipped to ~ 1.25D). As can be seen from the figure, the average velocity seen by the rear turbine is about 3.6 m/s, which is significantly lower than that seen by the front turbine.

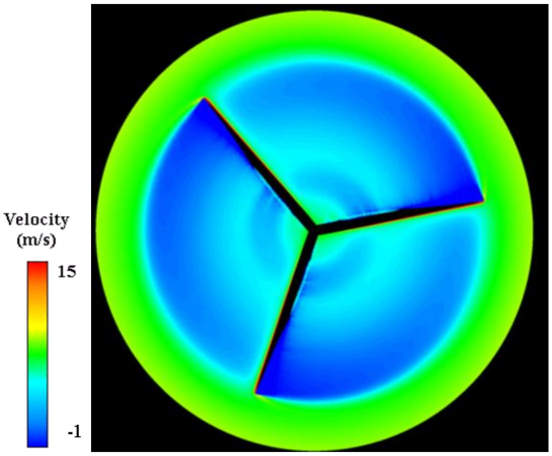


Figure 15 Steady flow field for rear turbine in tandem case (downstream distance = 7 diameters)

This is also borne out by looking at the axial velocity distribution along a cut down the midplane of the turbines (Figure 16). In this figure, one can see that there is significant negative velocity indicating that the flow field seen by the rear turbine is far from optimal.

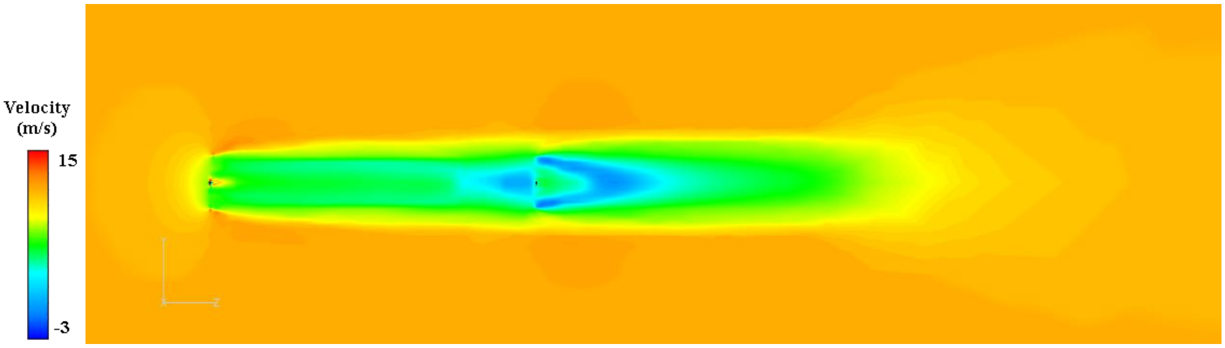


Figure 16 Steady velocity field for the tandem case (downstream distance = 7 rotor diameters)

Given the lack of power generation by the second turbine, it was decided to increase the streamwise spacing to 15 rotor diameters. In order to use the replication strategy, the single turbine case was rerun with a replication box extent of 13 diameters and the resulting solution was used to initialize the tandem case.A “from scratch” solution was also run on the tandem case for comparison purposes. The velocity field from the replication case, one diameter upstream of the rear turbine, is shown in Figure 17 with an average velocity of 5.5 m/s.

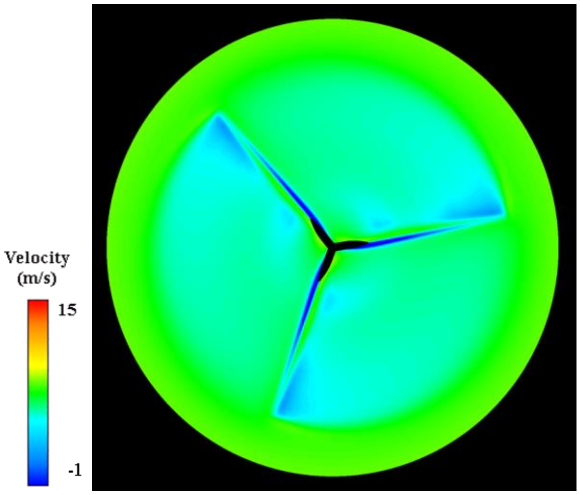


Figure 17 Steady flow field (clip-cut) entering rear turbine in the tandem case (downstream distance = 15 rotor diameters)

This is a significant improvement over the 7D case, where the average velocity was 3.6 m/s. This improvement is also evident in the axial velocity field along a cut down the midplane of the turbines (Figure 18).

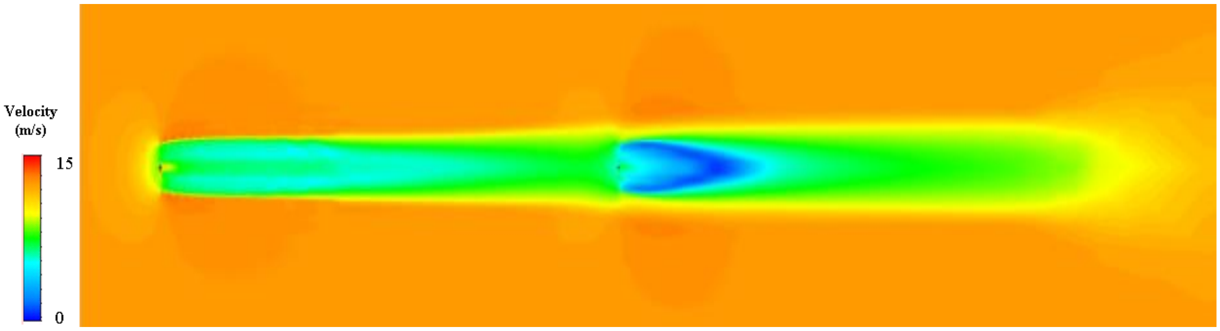


Figure 18 Steady velocity flow fields for the tandem case (streamwise distance = 15 rotor diameters)

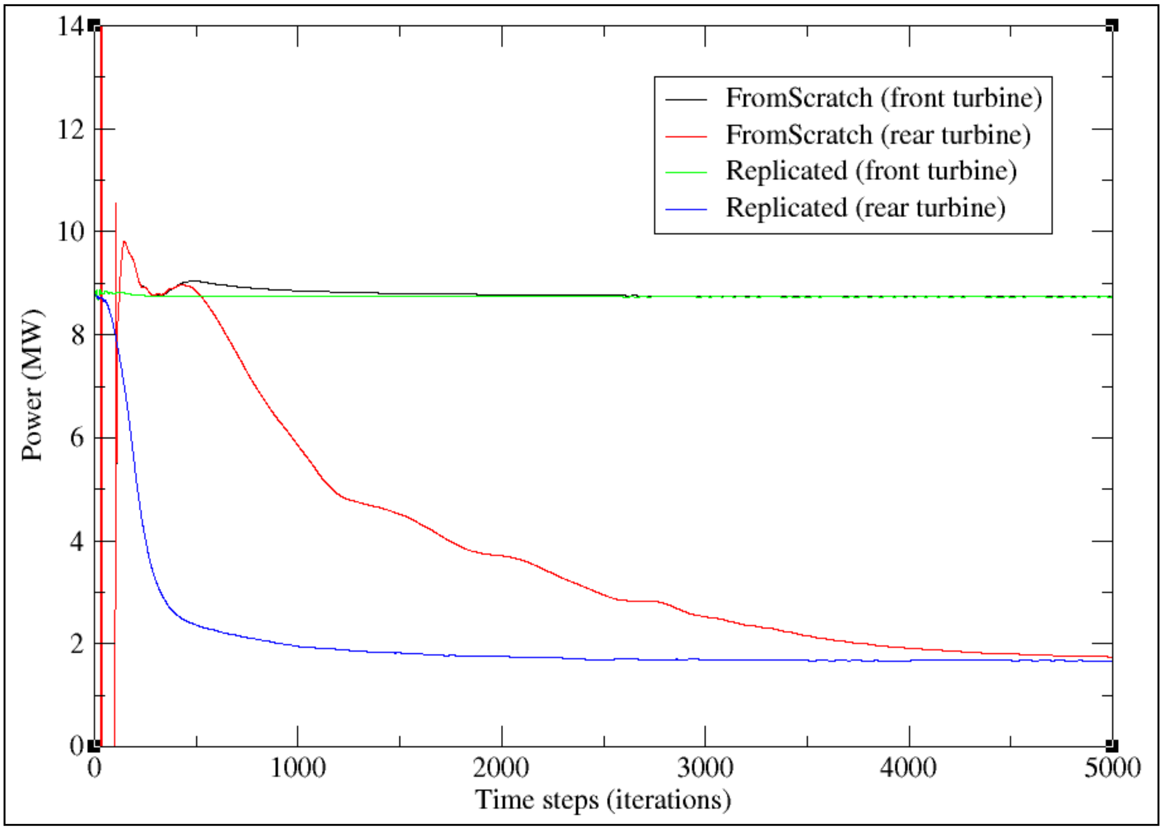


Figure 19 Steady power predictions for the different tandem cases (streamwise distance = 15 rotor diameters)

The resulting power production and a comparison to the “from scratch” case is shown in Figure 19. As can be seen, the rear turbine is now producing about 2 MW of power. Comparing the number of iterations required for convergence, it is clear that the replication strategy provides a significant advantage.

The replicated solution for the rear turbine has reached the converged value of power in about 2500 iterations, while the “from scratch” solution requires 5000 iterations. This is an improvement of 50% in terms of runtime.

Triangle Configuration

Two different triangle configurations were considered as part of this study. A lateral spacing of 2.5 and 5 rotor diameters were used, while the axial spacing was maintained at 15 rotor diameters. The resulting (steady) velocity fields for the 5D lateral separation are shown in Figure 20. As can be seen, there is no significant interaction between the turbines.

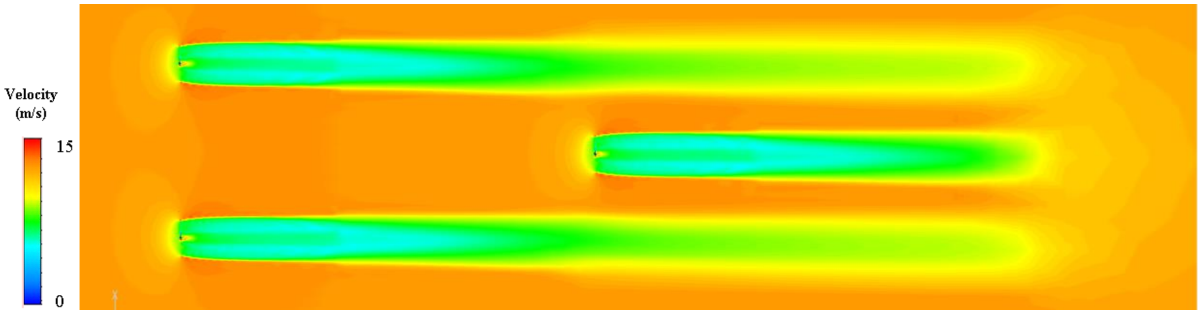


Figure 20 Steady velocity field for the triangle case (lateral distance = 5 rotor diameters)

This can also be seen in the power produced by the rear turbine (Figure 21). For this case, the power produced by the front and the rear turbines are nearly identical. Since there was no significant interaction between the turbines, this case was not run “from scratch”.

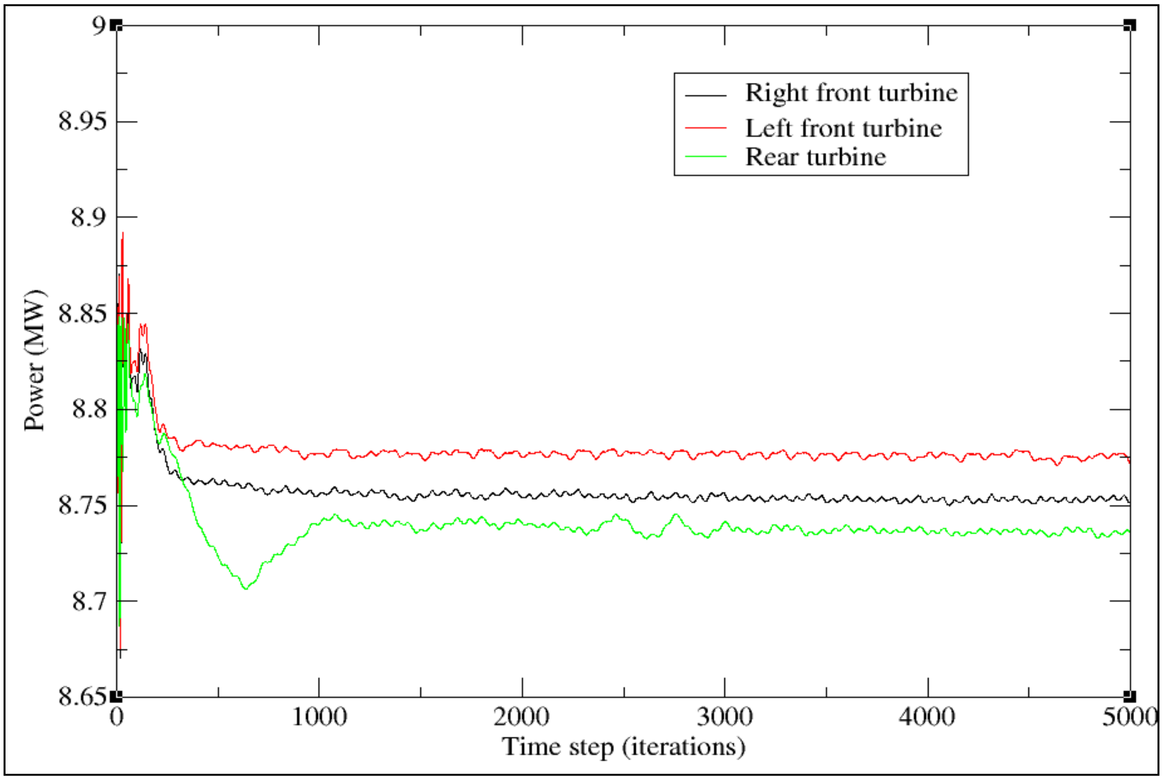


Figure 21 Steady power predictions for the triangle cases (lateral distances = 5 rotor diameters)

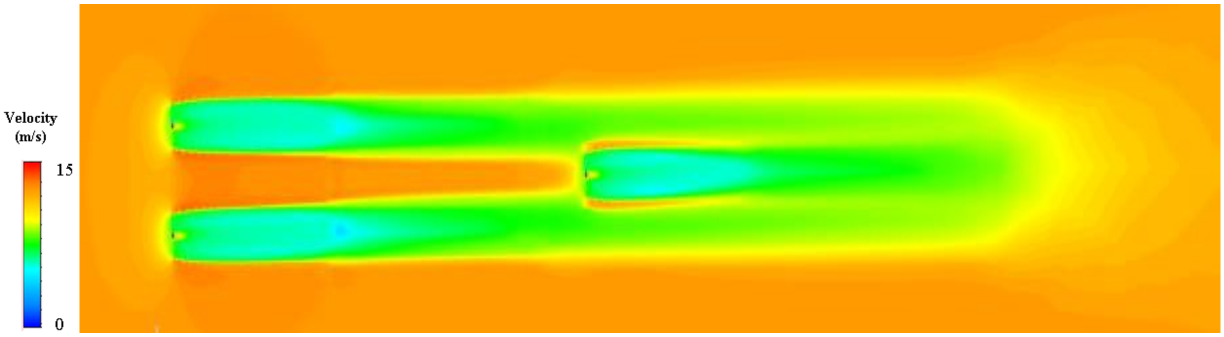


Figure 22 Steady velocity flow fields for the triangle case (2.5 D)

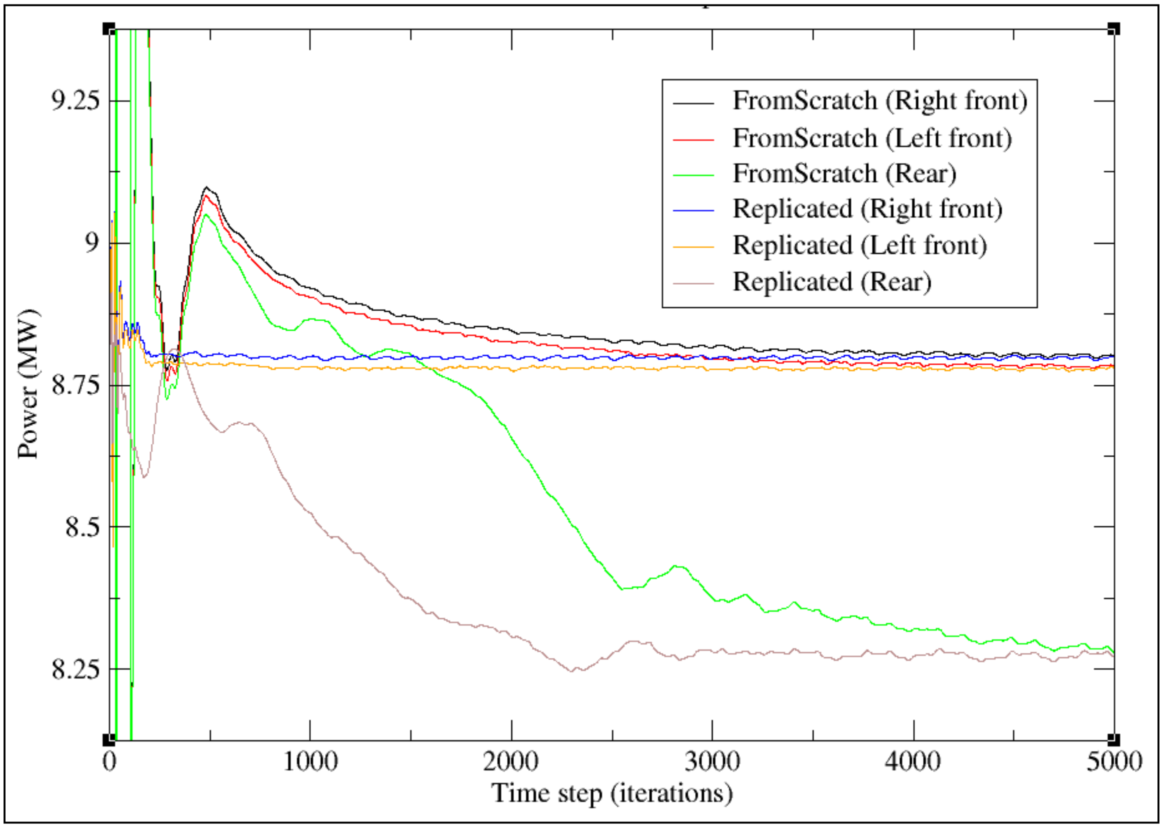


Figure 23 Steady power predictions for the triangle case (2.5 D)

The velocity field for the 2.5 D lateral spacing is shown in Figure 22. Comparing Figures 20 and 22, it is clear that the reduced lateral spacing results in significant interaction between the turbines. The wakes from the front turbines are on a trajectory to merge and can be seen impacting the rear turbine. This can be seen in the power produced (Figure 23). Figure 23 also includes the corresponding “from scratch” solution. The replicated solution for the rear turbine has reached the converged value of power in about 3000 iterations, while the “from scratch” solution requires 5000 iterations. This is an improvement of 40% in terms of runtime.

Unsteady runs

The idea is to obtain a time-accurate simulation of a single turbine and then replicate it for the tandem or triangle configurations and then iterate it in a time-accurate manner to convergence. However, the time scales involved in this computation were grossly underestimated. Consequently, only incomplete results are available for comparison for the tandem case.

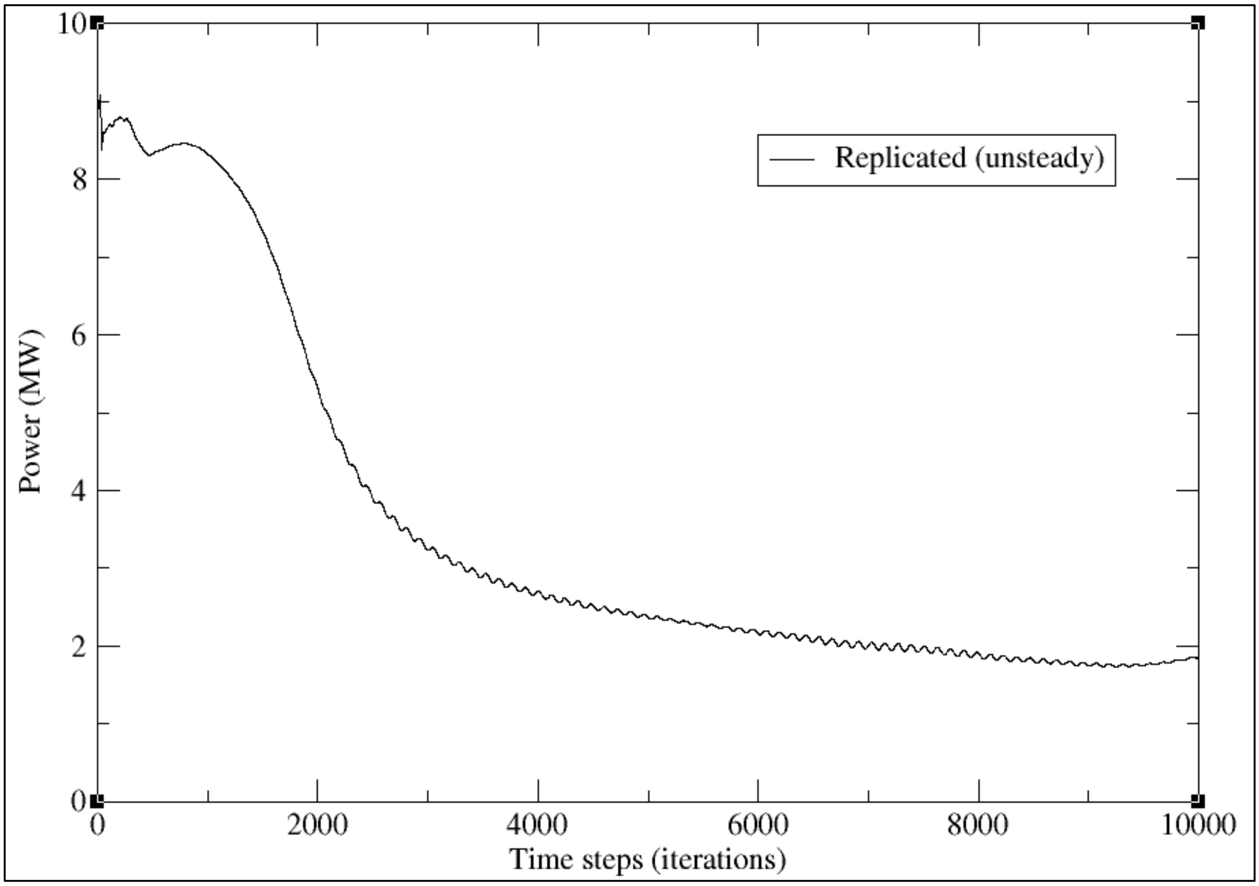


Figure 24 Power production of rear turbine for tandem case (15D; replicated; unsteady simulation for 10,000 steps)

An unsteady solution was obtained for the single turbine configuration with a replication box 13D long. This solution was obtained using a run that consisted of 5,000 steps of steady-state simulation followed by another 5,000 steps of unsteady simulation. This solution was replicated for the tandem case (15D separation) and the resulting flow field was advanced in a time accurate manner for 10,000 time steps at a time step of 0.01886 seconds. The power produced by the rear turbine is shown in Figure 24. As can be seen from the figure, the solution is nowhere near converged with significant variations still occurring at the end of 10,000 time steps.

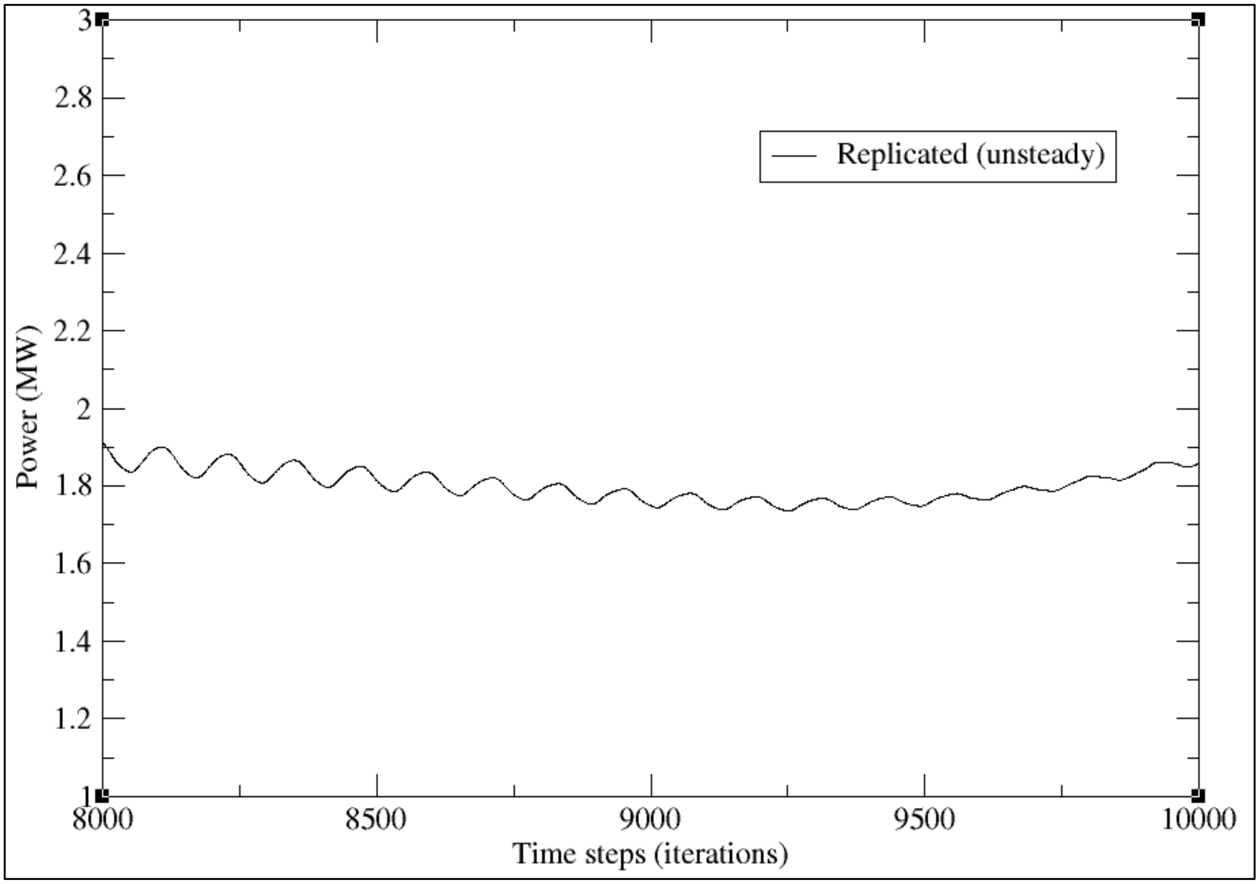


Figure 25 Zoomed-in power production of rear turbine for tandem case (15D; replicated; unsteady simulation for 10,000 steps)

In order to understand why this was happening, a simple analysis was undertaken. Assuming that the wake is convected at the freestream velocity, the number of time steps required for the wake to propagate the distance between the end of the replication box and the rear turbine (~360 m) is about 1,700 steps. However, the solution is not settled even after 10,000 steps. Further analysis points to the time-accurate simulation used to initialize the domain as the possible culprit. The single turbine case was run in a time accurate mode for 5,000 steps. However, the extent of the replication box was 13 D (~ 2300 m). This would have required the solution to be run time-accurate for almost 11,000 steps for the entire wake to be time accurate, i.e., there was a deficit of almost 6,000 steps in the single turbine time accurate simulation. When the unsteady simulation for the tandem configuration was started, the wake now took almost 7,700 steps to arrive (in a time-accurate fashion) at the rear turbine. At this point, the rear turbine started reacting to the time-accurate wake and this is the behavior that can be seen starting at around time step 9000 in Figure 24 and in Figure 25. It is possible that a further 4 to 5 revolutions (corresponding to an additional 1500 – 2000 steps) of time-accurate simulation would be required before the power production of the rear turbine would stabilize. Similar behavior was observed by researchers when running the BT2 test case, which involved model scale turbines operating in tandem. It was noticed that the rear turbines' power production was relatively unaffected at the beginning of the time-accurate simulation, but then it goes through a transient where its power production varies greatly and then stabilizes to a value that is different from the "steady" value. It is hypothesized that this transient is the "arrival" of the time-accurate wake of the front turbine and the rear turbine reacts to it over a certain period of time before attaining equilibrium. The scale of the BT2 test case was almost 3 orders of magnitude smaller than the current case, therefore, it was feasible to get a time accurate solution in a reasonable amount of time. Note that the replication technique was not used for the BT2 test case.

In an effort to speed up the replicated solution, the following trick was attempted. The solution was initialized (for the replication case) with the unsteady solution for a single turbine. It was then run in a "steady mode" for 1,000 time steps so that the interaction between the two turbines could be established. It was then switched over to an unsteady mode with 10 Newton subiterations and second order accuracy in time. The resulting power distribution is shown in Figure 26. As can be seen in the figure, the replicated (steady) solution, at the end of 1000 steps, appears to be settling down to a value of around 1.8 MW, which is close to the value obtained using the steady simulations. Once the time-accurate computations are started, the power produced jumps up to around 2.5 MW and seems to stabilize around that value. This tendency can be seen at the very end of the simulation in Figures 24 and 25, wherein the power value is also beginning to increase as a result of the interaction with the incoming wake.

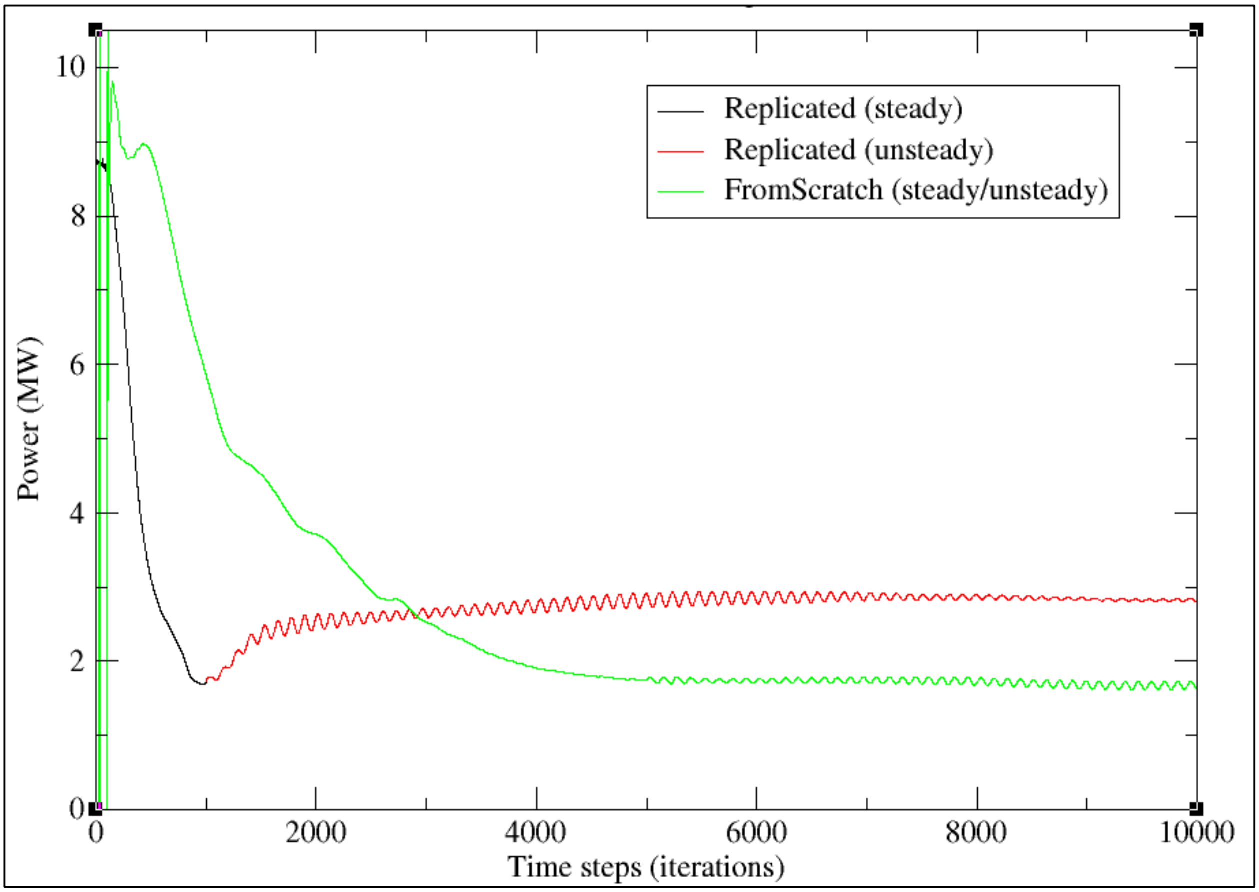


Figure 26 Time-accurate power predictions over the first 5000 time steps for the triangle case

Figure 26 also shows the power production for the “from scratch” simulation. The “from scratch” simulation was run steady for the first 5000 time steps and then switched over to a time accurate mode (10 Newton subiterations and second order accuracy in time) for another 5,000 time steps. Based on the earlier hypothesis, the “from scratch” simulation would have to be carried out for 13,000 time steps in order for the wake arriving at the rear turbine to be time-accurate, i.e., the simulation was about 8,000 steps short of achieving the required interaction. Consequently, the power production of the rear turbine stayed at the “steady state” value.

Based on these extrapolations, a comparison of the number of time steps required for the interaction to take place with and without replication can be made. For a spacing of 15D, the wake in the replicated case would have to travel a distance of 2D whereas in the “from scratch” case, it would have to travel the entire distance of 15D. At the time step that was used in the simulations, this would translate to a difference of about 11,300 time steps (about 1,700 for the replicated case assuming the wake at the end of the replication box was time-accurate and about 13,000 for the “from scratch” case). Given that the simulation would have to be carried out for another 2,000 time steps after the time-accurate interaction begins, the replication technique could potentially save almost 70% in computational cost.

CHAPTER VI

SUMMARY &CONCLUSIONS

A replication strategy aimed at reducing the computer time required for a blade-resolved wind farm simulation has been developed and implemented in *Tenasi*, a node-centered finite-volume flow solver. The replication strategy involves building meshes that have a replication box, i.e., a region that can be identified and the solution within which replicated as many times as required. This replication serves the purpose of initializing a flow field to better flow conditions as opposed to the traditional way of initializing the flow field to ones and zeroes.

Simulations of a representative wind farm consisting of 10 MW DTU wind turbines were carried out. The configurations considered included a tandem configuration and a triangle configuration. Several axial and lateral spacings were tested in order to find optimal operating points.

Cases were run using the replication strategy as well as “from scratch”. The replication strategy was found to be particularly effective for steady (not time-accurate) simulations of multiple turbines. The savings for the tandem case was around 50% while the savings for the triangle configuration was around 25%.

The unsteady simulations that have been carried out were not sufficient in order to fully ascertain the effectiveness of the replication technique. Itis hypothesized that the replication of the unsteady solution could save around the same order of magnitude (if not more) in computational cost as the steady cases.

REFERENCES

1. Outlook AE. with Projections to 2030. US Energy Information Administration. 2007.

2. De Vries O. Wind-tunnel tests on a model of a two-bladed horizontal-axis wind turbine and evaluation of an aerodynamic performance calculation method. Nationaal Lucht-en Ruimtevaartlaboratorium, 1979.

3. Alfredsson P, Dahlberg J. A preliminary wind tunnel study of windmill wake dispersion in various flow conditions, part 7. 1979.

4. Alfredsson P-H, Dahlberg J-A. Measurements of wake interaction effects on the power output from small wind turbine models. NASA STI/Recon Technical Report N. 1981;82:18720.

5. Kotb M, Haq A. Rigid wake model for a horizontal axis wind turbine. Wind Engineering. 1992;16(2):95-108.

6. Kotb M, Schetz J, editors. Windmill flowfield with nonuniform approach flow. European Wind Energy Conference; 1984.

7. Kotb M, Schetz J. Turbulent Flowfield Measurements Immediately Behind a Windmill Rotor in Uniform Flow. Three-dimensional Flow Phenomena in Fluid Machinery. 1985:87-94.

8. Vermeulen P, voor Toegepast-Natuurwetenschappelijk NCO. A wind tunnel study of the wake of a horizontal axis wind turbine: TNO; 1978.

9. Vermeulen P, voor Toegepast-Natuurwetenschappelijk NCO. Studies of the wake structure of model wind turbine generators: TNO; 1979.

10. Madsen HA, Larsen GC, Thomsen K. Wake flow characteristics in low ambient turbulence conditions. Proceedings of the Copenhagen Offshore Wind. 2005.

11. Thomsen K, Madsen HA, Larsen GC. A new method can predict detailed response for turbines in wind farms. Fact sheet AED-RB-16 (EN), Risoe National Laboratory, Roskilde, Denmark. 2003.

12. Bingöl F, Mann J, Larsen GC. Laser measurements of wake dynamics. EWEC2007, Milan, Italy. 2007.

13. Medici D. Experimental studies of wind turbine wakes: power optimisation and meandering. 2005.

14. Mikkelsen R. Actuator disc methods applied to wind turbines: Technical University of Denmark; 2003.

15. Sørensen J, Shen W, Munduate X. Analysis of wake states by a full-field actuator disc model. Wind Energy. 1998;1(2):73-88.

16. Sørensen JN, Mikkelsen R, Troldborg N. Simulation and modelling of turbulence in wind farms. Proceedings EWEC 2007. 2007.

17. Castellani F, Gravdahl A, Crasto G, Piccioni E, Vignaroli A. A Practical Approach in the CFD Simulation of Off-shore Wind Farms through the Actuator Disc Technique. Energy Procedia. 2013;35:274-84.

18. Martınez LA, Leonardi S, Churchfield MJ, Moriarty PJ. A comparison of actuator disk and actuator line wind turbine models and best practices for their use. AIAA Paper. 2012(2012-0900).

19. Churchfield MJ, Lee S, Moriarty PJ, Martinez LA, Leonardi S, Vijayakumar G, et al. A large-eddy simulation of wind-plant aerodynamics. AIAA paper. 2012;537.

20. Churchfield MJ, Wang Z, Schmitz S. Modeling Wind Turbine Tower and Nacelle Effects within an Actuator Line Model.

21. Zahle F, Johansen J, Sørensen N, editors. Wind turbine aerodynamics using an incompressible overset grid method. European Wind Energy Conference & Exhibition, EWEC 2007; 2007.

22. Troldborg N, Zahle F, Réthoré P-E, Sørensen NN, editors. Comparison of the wake of different types of wind turbine CFD models. 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition; 2012.

23. Sitaraman J, Mavriplis D, Duque EP, editors. Wind farm simulations using a full rotor model for wind turbines. Proceedings of the AIAA SciTech 2014 Meeting, National Harbor, MD; 2014.

24. Henrick AK, Aslam TD, Powers JM. Mapped weighted essentially non-oscillatory schemes: achieving optimal order near critical points. Journal of Computational Physics. 2005;207(2):542-67.

25. Titarev V, Toro E. Finite-volume WENO schemes for three-dimensional conservation laws. Journal of Computational Physics. 2004;201(1):238-60.

26. Hyams DG. An investigation of parallel implicit solution algorithms for incompressible flows on unstructured topologies2000.

VITA

Walied E. Hassan was born in Khartoum, Sudan. He attended Comboni Elementary and continued to Elsheikh Mustafa Elamin High School in Khartoum, Sudan. After graduation, attended the University of Khartoum where he became interested in Mechanical Engineering. Walied completed an International Study Program with the Islamic university in Malaysia, which was the impetus for him to continue his education. He completed the Bachelors of Science degree in August 2009 in Mechanical Engineering. Walied worked for three years for Asawer Oil & Gas co. before accepting a graduate research assistantship at the University of Tennessee at Chattanooga in the Computational Engineering Program. Walied graduated with a Master of Science: Engineering in December 2015.