Matthew A. Lackner, Ph.D.
UMASS Wind Energy Center

## Graphical Processing Units offer a promising solution for parallelizing computations of N-body problems and drastically accelerating computations.

UNSTEADY AERODYNAMICS CAUSED by platform kinematics represents a significant increase in system complexity for offshore wind turbines. Offshore floating wind turbines (OFWTs) present significant advantages over conventional offshore fixed foundation wind turbines, and can harness the vast deep water wind resource while avoiding many of the public acceptance issues that have impeded near shore development. OFWTs also pose significant challenges that are not present for conventional offshore technology, most critically the additional dynamic behavior of the floating platform.

In a previous article in this space by Lackner and Sebastian (a former PhD student at UMass), the additional complexity of the aerodynamics for OFWTs was highlighted. In particular, a variety of analyses revealed that OFWTs have a greater fraction of unsteady flow energy due to platform kinematics than a comparable monopile system, momentum balance assumptions that underpin all blade-element momentum and generalized dynamic wake-based analysis break down more often for OFWTs than for monopoles, and transition states are more prevalent.

As a compromise between the computational complexity of computational fluid dynamics (CFD), and the limited applicability of Blade Element Momentum Theory (BEM) to complex flowfields, a potential flow method has been chosen by researchers at UMass Amherst to model the aerodynamics of OFWTs. Time-marching free vortex wake methods (FVMs), a subset of potential flow, numerically advect the wake lattice, which is composed of Lagrangian markers connected by vortex filaments. This approach has been used for a number of decades, in particular in rotorcraft aerodynamic analysis. Recognizing this, Sebastian and Lackner developed the Wake Induced Dynamics Simulator (WInDS) code, a lifting-line theory (LLT) based FVM developed for OFWTs and validated via comparison to analytical models and experimental data. The results of the wake development of this model compare favorably to both the MEXICO experiment by the Energy research Centre of the Netherlands (ECN) and a two bladed experiment by the Delft University of Technology (DUT). Comprehensive analyses of the aerodynamics of three floating platform models – a spar buoy, tension leg platform, and barge – were conducted and the complexity of the flow field was highlight.

However, one of the major challenges in this initial research was the computational cost of the FVM calculations using WInDS, which limited the feasible spatial and temporal discretization of the wake, and thus the accuracy of the solution and the ability to conduct a large number of simulations. The main computational cost of the model lies in the solution of the N-body Biot-Savart Law, which is used to calculate the induced velocity at all Lagrangian markers due to all the vortex filaments in the domain. The straightforward "brute-force" solution of the N-Body problem can be prohibitively slow for large numbers of vortex filaments, i.e. large values of N, as the solution time is proportional to $N^2$. A typical wind turbine simulation of 60 seconds can have values of N of nearly 100,000. To address this problem and accelerate the computational speed of WInDS, Lackner and deVelder (a PhD student at UMass) have explored a parallel computing approach to the Biot-Savart Law. Specifically they investigated parallelism using a low cost, off-the-shelf, Fermi based graphics processing unit (GPU) with both a "naive" and tiled shared-memory implementation of the Compute Unified Device Architecture (CUDA) kernel. The Biot-Savart Law was coded in CUDA and then compiled as a "mex" function, which can be called by Matlab, which is the language that WInDS is written in. In this way, the simplicity of developing the overall code in Matlab is maintained, while the main computational cost of the code is solved outside of Matlab on the GPU. The results have been promising. Identical 30-second simulations were conducted, solved either with the standard CPU approach or on the GPU. The GPU implementation decreased the total computation time by a factor of 25 times. These results open the door for more complex FVM calculations with WInDS with higher levels of discretization and the possibility to conduct design optimization. Future work will investigate even greater computational gains using the Barnes and Hut tree-code (BHTC) as a low-barrier-to-entry option for algorithmic improvement, taking the computational expense from $O(N^2)$ to O(NlogN).

Matthew A. Lackner, Ph.D., is an assistant professor of mechanical and industrial engineering at the University of Massachusetts Amherst and a member of its Wind Energy Center. He can be reached at (413) 545-4713 or lackner@ecs.umass.edu. Also visit www.umass.edu/windenergy.