



Full-System Linearization for Floating Offshore Wind Turbines in OpenFAST

Preprint

Jason M. Jonkman, Alan D. Wright, Greg J. Hayman,
and Amy N. Robertson

National Renewable Energy Laboratory

*Presented at the ASME 2018 1st International
Offshore Wind Technical Conference
San Francisco, California
November 4-7, 2018*

**NREL is a national laboratory of the U.S. Department of Energy
Office of Energy Efficiency & Renewable Energy
Operated by the Alliance for Sustainable Energy, LLC**

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at www.nrel.gov/publications.

Contract No. DE-AC36-08GO28308

Conference Paper
NREL/CP-5000-71865
December 2018



Full-System Linearization for Floating Offshore Wind Turbines in OpenFAST

Preprint

Jason M. Jonkman, Alan D. Wright, Greg J. Hayman,
and Amy N. Robertson

National Renewable Energy Laboratory

Suggested Citation

Jonkman, Jason, Alan Wright, Greg Hayman, and Amy Robertson. 2018. *Full-System Linearization for Floating Offshore Wind Turbines in OpenFAST: Preprint*. Golden, CO: National Renewable Energy Laboratory. NREL/CP-5000-71865.
<https://www.nrel.gov/docs/fy19osti/71865.pdf>

**NREL is a national laboratory of the U.S. Department of Energy
Office of Energy Efficiency & Renewable Energy
Operated by the Alliance for Sustainable Energy, LLC**

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at www.nrel.gov/publications.

Contract No. DE-AC36-08GO28308

Conference Paper
NREL/CP-5000-71865
December 2018

National Renewable Energy Laboratory
15013 Denver West Parkway
Golden, CO 80401
303-275-3000 • www.nrel.gov

NOTICE

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Wind Energy Technologies Office. The views expressed herein do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at www.nrel.gov/publications.

U.S. Department of Energy (DOE) reports produced after 1991 and a growing number of pre-1991 documents are available free via www.OSTI.gov.

Cover Photos by Dennis Schroeder: (clockwise, left to right) NREL 51934, NREL 45897, NREL 42160, NREL 45891, NREL 48097, NREL 46526.

NREL prints on paper that contains recycled content.

FULL-SYSTEM LINEARIZATION FOR FLOATING OFFSHORE WIND TURBINES IN OPENFAST

Jason M Jonkman, Alan D Wright, Greg J Hayman, Amy N Robertson
National Renewable Energy Laboratory
15013 Denver West Parkway
Golden, Colorado 80401 (USA)

ABSTRACT

The wind engineering community relies on multiphysics engineering software to run nonlinear time-domain simulations (e.g., for design-standards-based loads analysis). Although most physics involved in wind energy are nonlinear, linearization of the underlying nonlinear system equations is often advantageous to understand the system properties and exploit well-established methods and tools for analyzing linear systems. This paper presents the development of the new linearization functionality of the open-source engineering tool OpenFAST for floating offshore wind turbines, as well as the concepts and mathematical background needed to understand and apply it.

INTRODUCTION

To support design and analysis—so that wind turbines are innovative, optimized, reliable, and cost-effective—the wind industry and research communities rely on engineering software (i.e., design tools) capable of predicting the coupled dynamic loads and responses of the wind system. OpenFAST (formerly known as FAST), developed by the National Renewable Energy Laboratory (NREL) via support from the U.S. Department of Energy, is an open-source multiphysics tool that enables the design of wind turbines [1]. OpenFAST models the important physical phenomena and system couplings, including the environmental excitation (wind, waves, and current) and full-system dynamic response (rotor, drivetrain, nacelle, support structure, and controller) under both normal (for fatigue) and extreme (for ultimate) loading conditions. The OpenFAST model enables the analysis of a range of wind turbine configurations, including two- or three-bladed horizontal-axis rotors, pitch or stall regulation, rigid or teetering hub, upwind or downwind rotor, and lattice or tubular towers. The wind turbine can be modeled on land or offshore on bottom-fixed or floating substructures.

The primary use of OpenFAST is to run nonlinear time-domain simulations (e.g., for design-standards-based loads analysis). Although most physics involved in wind energy are nonlinear, linearization of the underlying nonlinear system equations is often advantageous to understand the system response and exploit well-established methods and tools for analyzing linear systems. For example, linear state-space

models can be transformed to transfer functions, impulse-response functions, or frequency-response functions. The ability to generate linearized models is important for eigenanalysis (to derive structural natural frequencies, damping ratios, and mode shapes), controls design (based on linear state-space models), stability analysis, gradients for optimization problems, and support for the development of reduced-order models.

Previous FAST linearization work focused on 1) structuring the FAST v8 source code to enable linearization (FAST v8 is the predecessor of OpenFAST); 2) developing the general approach to linearizing the mesh mapping within the module-to-module, input-output coupling relationships, including rotations; 3) linearizing core (but not all) features of the land-based modules of FAST (InflowWind, AeroDyn, ServoDyn, and ElastoDyn) and their coupling; and 4) verifying this implementation by applying the tool to sample cases [2]. More recently, the ability to linearize OpenFAST models with advanced structural dynamics for aeroelastically tailored blades using the BeamDyn module was introduced and verified [3]. This work extends these efforts to linearizing core (but not all) features of the floating offshore wind turbine (FOWT) modules in OpenFAST and their coupling, including the hydrodynamics in HydroDyn and moorings in MAP++, and verifying the updated implementation.¹

The HydroDyn hydrodynamics module of OpenFAST [1] allows for multiple approaches for calculating the hydrodynamic loads on floating platforms, including a potential-flow theory solution, a strip-theory solution, or a hybrid combination of the two. Waves may be regular (periodic) or irregular (stochastic), and long crested (unidirectional) or short crested (with wave energy spread across a range of directions). Second-order terms in the wave kinematics and/or potential-flow solution are available options. The MAP++ module of OpenFAST [1] models mooring systems quasi-statically, considering geometric nonlinearities of catenary or taut lines, elastic stretching, line weight, and buoyancy as well as clump weights and buoyancy tanks, line-to-line interconnections, and seabed friction.

¹The offshore linearization functionality of older versions of FAST (v6 through v7) was limited to structural linearization only, requiring disabling or assuming a quasi-static treatment of other physics, and linearization of wave-excitation and wave-radiation loads was not possible.

The overall linearization approach that the FAST modularization framework was designed to support is explained in [4] and is consistent with the present implementation. Details on the linearization of the land-based modules and their coupling, including mesh mapping, is explained in [2,3]. Without replicating most of the information, this paper uses the same approach and nomenclature of [2,3,4], adding details about the linearization of the floating offshore modules HydroDyn and MAP++ and their coupling, including mesh mapping.

The mathematical background presented in this paper is currently being implemented in the OpenFAST source code. Unfortunately, the implementation at the time of this writing has not yet been completed enough to produce results. Results will be presented in future work to highlight the functionality and verify the implementation (e.g., for a sample case whereby the natural frequencies and damping of the NREL 5-MW baseline wind turbine atop a spar buoy will be calculated as a function of rotational speed and presented in Campbell-diagram form).

The linearization of OpenFAST involves: 1) finding an operating point (OP), 2) linearizing the underlying nonlinear equations of each module about the OP, 3) linearizing the module-to-module input-output coupling relationships in the OpenFAST glue code about the OP, and 4) combining all linearized matrices into the full-system linear state-space model and exporting those matrices and the OP to a file. Each step is highlighted in the following sections.

OPERATING-POINT DETERMINATION

OP (or fixed-point) determination is an important first step in the linearization process because a linear representation of a nonlinear system is only valid for small deviations (perturbations) from an OP. In the current release of OpenFAST, an OP can be defined by given initial conditions (time zero) or a given time (or times) in the nonlinear time-marching process. It is usually important for the OP to be a static-equilibrium condition (for parked/idling turbines) or steady-state condition (for operating turbines); otherwise, it may have an undesirable effect on the linear system matrices.

An OP is defined by given values for the continuous-time states, $x|_{op}$, discrete-time states, $x^d|_{op}$, inputs, $u|_{op}$, and time, $t|_{op}$, for each module. Equations (1a), (1c), and (1d) from [4] can then be used to calculate the OP values of the first time derivative of the continuous-time states, $\dot{x}|_{op}$, constraint (algebraic) states, $z|_{op}$, and outputs, $y|_{op}$, for each module. Each of these variables can be perturbed (represented by Δ) about their respective OP values as given by Eq. (11) from [4] (e.g., for module inputs $u = u|_{op} + \Delta u$). References [2,3] clarify how this operation is extended to rotations (orientations) in three dimensions (3D), which do not reside in a linear space.

The number of states, inputs, and outputs (i.e., the size of the vectors x , x^d , z , u , and y) depend on the features enabled in OpenFAST.

MODULE LINEARIZATION

As explained in [4], the FAST modularization framework supports a very general (need not be linear) state-space formulation, with any combination of continuous-time-state, discrete-time-state, constraint- (algebraic-) state, other- (e.g., logical) state, and output equations. However, for a module to support linearization, the formulation is limited to a hybrid semiexplicit differential-algebraic equation (DAE) of index 1,² which has the following limitations: 1) the continuous-time-state derivatives and discrete-time-state updates must be written as an explicit function of the states, inputs, and parameters; 2) the constraints must be of index 1; and 3) other states are used only for time-integration or when acting as parameters in the linearization process.

To support linearization, a module must also be able to export Jacobian matrices for the state and output equations with respect to the states and inputs. The OpenFAST module states, inputs, and outputs kept in the linearization process for the features linearized to date are summarized in Table 1. The OpenFAST module features linearized to date include only continuous-time and constraint states (no features with discrete-time states have yet been linearized).

The linearized form of a general module is given by Eqs. (12) and (13) from [4]; the simplified forms for the land-based modules InflowWind, AeroDyn, ServoDyn, ElastoDyn, and BeamDyn are given in [2,3]. The simplified forms for the newly linearized offshore modules HydroDyn and MAP++ are given next, along with a description of how each module is linearized.

²Indeed, linearization is not possible or desirable for all models, but only those limited to this restricted form. Most models are simplifications of this most general form (e.g., for a module without discrete-time and constraint states, the continuous-time-state equations form ordinary differential equations).

TABLE 1. MODULE STATES, INPUTS, AND OUTPUTS KEPT IN THE OPENFAST LINEARIZATION PROCESS TO DATE.

Module	States	Inputs	Outputs
InflowWind (IfW)	<ul style="list-style-type: none"> None 	<ul style="list-style-type: none"> Positions where the undisturbed (inflow) wind will be output Disturbances of horizontal wind speed, power-law shear exponent, and wind-propagation direction 	<ul style="list-style-type: none"> Undisturbed (inflow) wind velocity at input positions User-selected wind-inflow outputs
ServoDyn (SrvD)	<ul style="list-style-type: none"> None 	<ul style="list-style-type: none"> Nacelle-yaw angle and rate Generator speed 	<ul style="list-style-type: none"> Blade-pitch-angle command (independent) Nacelle-yaw moment Generator torque and electrical power User-selected control and electrical-drive outputs
ElastoDyn (ED)	<ul style="list-style-type: none"> Position and orientation and translational and rotational velocities of the floating platform (continuous states) Relative bending-mode degrees of freedom of the blades and tower and their first-time derivatives (continuous states) Relative rotational displacement and velocity of the nacelle-yaw, generator azimuth, shaft torsion, and rotor-teeter (continuous states) 	<ul style="list-style-type: none"> Applied point forces and moments distributed along the blades and tower Applied point forces and moments lumped on the hub, nacelle, and platform Blade-pitch-angle command (both independent and rotor-collective) Nacelle-yaw moment Generator torque 	<ul style="list-style-type: none"> Translational displacements, orientations, translational and rotational velocities, and translational and rotation accelerations of analysis nodes along the blades and tower Translational displacements, orientations, translational and rotation accelerations of the blade-root, nacelle, and platform reference points Translational displacement, orientation, and rotational velocity of the hub reference point Nacelle-yaw angle and rate Generator speed User-selected structural outputs
BeamDyn (BD)	<ul style="list-style-type: none"> Positions and orientations and translational and rotational velocities of nodes distributed along each blade (continuous states) 	<ul style="list-style-type: none"> Translational displacements, orientations, translational and rotational velocities, and translational and rotational accelerations of each blade root Applied point forces and moments distributed along each blade Applied line (per-unit length) forces and moments distributed along each blade 	<ul style="list-style-type: none"> Reaction point forces and moments lumped at each blade root Translational displacements, orientations, translational and rotational velocities, and translational and rotation accelerations of nodes along each blade User-selected structural outputs
AeroDyn (AD)	<ul style="list-style-type: none"> Inflow angle at each blade analysis node/airfoil (constraint states) 	<ul style="list-style-type: none"> Translational displacements, orientations, and translational velocities of analysis nodes along the blades and tower Orientation of the blade-root reference point Translational displacement, orientation, and rotational velocity of the hub reference point Undisturbed (inflow) wind velocities at analysis nodes along the blades and tower 	<ul style="list-style-type: none"> Aerodynamic-applied line (per-unit length) forces and moments distributed along the blades and tower User-selected aerodynamic outputs
HydroDyn (HD)	<ul style="list-style-type: none"> State-space-based wave-excitation states (continuous states) State-space-based wave-radiation states (continuous states) 	<ul style="list-style-type: none"> Translational displacements, orientations, translational and rotational velocities, and translational and rotational accelerations of analysis nodes distributed along the floating platform Translational displacement, orientation, translation and rotational velocities, and translational and rotational accelerations of the platform reference point Disturbance of wave elevation at the platform reference point 	<ul style="list-style-type: none"> Hydrodynamic-applied point forces and moments distributed along the floating platform Hydrodynamic-applied line (per-unit length) forces and moments distributed along the floating platform Hydrodynamic-applied point force and moment at the platform reference point User-selected hydrodynamic outputs
MAP++ (MAP)	<ul style="list-style-type: none"> Horizontal and vertical tensions at the fairlead of each mooring line (constraint states) Positions of each connect node (constraint states for multi-segmented mooring) 	<ul style="list-style-type: none"> Translational displacements of each fairlead 	<ul style="list-style-type: none"> Reaction point forces (tensions) lumped at each fairlead User-selected mooring outputs

The linearization of the HydroDyn (HD) module applies to both the strip-theory solution, potential-flow solution, or a hybrid combination of the two. Linear state-space-based wave-excitation and wave-radiation models have been added to HydroDyn to enable linearization of the potential-flow solution, including wave-excitation loads (with diffraction) from disturbances of wave elevation and wave-radiation loads with free-surface memory effects.

The linear state-space-based wave-radiation model was previously available in HydroDyn within FAST v8. As shown in [5], the model approximates the convolution term in the Cummins equation as shown in Eq. (1), with a linear state-space model as shown in Eq. (2), where F_{Rdm} is the wave-radiation loads; K_{Rdm} is the wave-radiation-retardation kernels; \dot{q}_{Pifm} is the floating platform translation and rotational velocities; x_{Rdm} are the wave-radiation states; and A_{Rdm} ,

B_{Rdm} , and C_{Rdm} are the matrices of the linear state-space wave-radiation model. The more wave-radiation states there are, the better the state-space model can approximate the convolution. Within HydroDyn, the user can select either the convolution-based wave-radiation model, which is implemented via numerical convolution using discrete-time states (which, if linearized, would overly complicate application of the linear state-space model), or the linear state-space-based wave-radiation model, which is implemented using matrices derived from the SS_Fitting preprocessor [6] or equivalent.

$$F_{Rdm}(t) = -\int_0^t K_{Rdm}(t-\tau) \dot{q}_{Ptfm}(\tau) d\tau \quad (1)$$

$$\dot{x}_{Rdm} = A_{Rdm} x_{Rdm} + B_{Rdm} \dot{q}_{Ptfm} \quad (2a)$$

$$F_{Rdm} \cong C_{Rdm} x_{Rdm} \quad (2b)$$

The linear state-space-based wave-excitation model was newly introduced in HydroDyn within OpenFAST with the addition of the linearization functionality for FOWTs. Similar to the linear state-space-based wave-radiation model, the model approximates the infinite integral shown in Eq. (3) with a linear state-space model as shown in Eq. (4), where F_{Excitn} is the wave-excitation loads under unidirectional waves; K_{Excitn} is the incident wave-excitation kernels; ζ is the wave elevation; ζ_c is a time-shifted wave elevation; x_{Excitn} is the wave-excitation states; and A_{Excitn} , B_{Excitn} , and C_{Excitn} are the matrices of the linear state-space wave-excitation model. Further details on this new linear state-space-based wave-excitation model are provided in Annex A. Within HydroDyn, the user can now select either a Fourier-transform-based wave-excitation model, which is implemented with discrete Fourier transforms (which is not conducive to linearization), or the new linear state-space-based wave-excitation model.

$$F_{Excitn}(t) = \int_{-\infty}^{\infty} K_{Excitn}(t-\tau) \zeta(\tau) d\tau \quad (3)$$

$$\dot{x}_{Excitn} = A_{Excitn} x_{Excitn} + B_{Excitn} \zeta_c \quad (4a)$$

$$F_{Excitn} \cong C_{Excitn} x_{Excitn} \quad (4b)$$

When linear state-space-based wave excitation or wave radiation are enabled in the potential-flow solution, continuous-time states associated with the excitation and radiation damping are included in the linearized HydroDyn model. Linearization is permitted only in still water ($\zeta_c|_{op} = x_{Excitn}|_{op} = F_{Excitn}|_{op} = 0$)³ and is not permitted when

³The restriction that linearization is permitted only in still water is in place to ensure that a static-equilibrium condition (for parked/idling turbines) or steady-state condition (for operating turbines) can be found for the OP (the

Fourier-transform-based wave-excitation, convolution-based wave-radiation, wave directional spreading, second-order wave kinematics, or second-order potential-flow loads are enabled. The linearized form of the HydroDyn state and output

equations is given by Eq. (5), where $\Delta x^{(HD)} = \begin{Bmatrix} \Delta x_{Excitn} \\ \Delta x_{Rdm} \end{Bmatrix}$,

$\Delta u^{(HD)}$ includes contributions from $\Delta \zeta$ and $\Delta \dot{q}_{Ptfm}$, and

$\Delta y^{(HD)}$ includes contributions from ΔF_{Excitn} and ΔF_{Rdm} .

$$\Delta \dot{x}^{(HD)} = A^{(HD)} \Delta x^{(HD)} + B^{(HD)} \Delta u^{(HD)} \quad (5a)$$

$$\Delta y^{(HD)} = C^{(HD)} \Delta x^{(HD)} + D^{(HD)} \Delta u^{(HD)} \quad (5b)$$

The continuous-state matrix, input matrix, continuous-state output matrix, and the input-transmission matrix are the Jacobians of the state and output equations relative to the states and inputs about the OP, computed analytically for the Jacobians of the state equations and numerically via a central-difference perturbation technique for the Jacobians of the output equations as shown in Eq. (6), where $X^{(HD)}$ is the

continuous-state functions and $Y^{(HD)}$ is the output functions of HydroDyn. For inputs that are rotations in 3D (i.e., $u = A$ and $\Delta u = \Delta \bar{\theta}$), it is implied that $u|_{op}^{(HD)} + \Delta u^{(HD)}$, $u|_{op}^{(HD)} - \Delta u^{(HD)}$,

and $2\Delta u^{(HD)}$ are written as $A|_{op}^{(HD)} f_{\Delta A}(\Delta \bar{\theta}^{(HD)})$,

$A|_{op}^{(HD)} f_{\Delta A}(-\Delta \bar{\theta}^{(HD)})$, and $2\Delta \bar{\theta}^{(HD)}$, respectively. In Eq.

(6b), the column order is dictated by the order of HydroDyn inputs, and 0 are appropriately sized zero matrices. For the numerically computed Jacobians, the default perturbation sizes are hard-coded within HydroDyn (but can be customized by recompiling) and are ≈ 0.035 m for translational states, 2° for rotational states and inputs, and fractions of the water depth for translational inputs. The Jacobians contain the linearized contributions from 1) state-space-based wave excitation, 2) hydrodynamic added mass, 3) state-space-based wave-radiation damping, 4) hydrostatic restoring, and 5) linearized viscous drag. The latter is found by numerically linearizing the viscous-drag term from the relative form of Morison's equation in the strip-theory solution (plus member-end effects) as in Eq. (6), not based on a stochastic linearization method (e.g., [7]).

time-domain solution could not reach a static-equilibrium or steady-state condition if waves were present). This implies that the OP values of the wave elevation, wave-excitation states, and wave-excitation loads are zero. That said, when linear state-space-based wave-excitation is enabled in the potential-flow solution, the linear system matrices calculated and output by OpenFAST include the contributions from perturbations of the wave elevation (i.e., $\Delta \zeta$ need not be zero), and so, even though the OP is determined in still water, the influence of wave-excitation may still be included.

$$A^{(HD)} = \frac{\partial X}{\partial x} \Big|_{op}^{(HD)} = \begin{bmatrix} A_{ExcIn} & 0 \\ 0 & A_{Rdm} \end{bmatrix} \quad (6a)$$

$$B^{(HD)} = \frac{\partial X}{\partial u} \Big|_{op}^{(HD)} = \begin{bmatrix} 0 & 0 & 0 & B_{ExcIn} \\ 0 & B_{Rdm} & 0 & 0 \end{bmatrix} \quad (6b)$$

$$C^{(HD)} = \frac{\partial Y}{\partial x} \Big|_{op}^{(HD)} = \frac{Y(x|_{op} + \Delta x, u|_{op}, t|_{op}) - Y(x|_{op} - \Delta x, u|_{op}, t|_{op})}{2\Delta x} \Big|_{op}^{(HD)} \quad (6c)$$

$$D^{(HD)} = \frac{\partial Y}{\partial u} \Big|_{op}^{(HD)} = \frac{Y(x|_{op}, u|_{op} + \Delta u, t|_{op}) - Y(x|_{op}, u|_{op} - \Delta u, t|_{op})}{2\Delta u} \Big|_{op}^{(HD)} \quad (6d)$$

The MAP++ (MAP) module has constraint (algebraic) states, there are no restrictions to linearization, and the linearized form is given by $\Delta y^{(MAP)} = D^{(MAP)} \Delta u^{(MAP)}$. The input-transmission matrix is the Jacobian of the output equations relative to the inputs about the OP, computed numerically via a central-difference perturbation technique as shown in Eq. (7), where $Y^{(MAP)}$ is the output functions of MAP++. For inputs that are rotations in 3D (i.e., $u = \Lambda$ and $\Delta u = \Delta \bar{\theta}$), it is implied that $u|_{op}^{(MAP)} + \Delta u^{(MAP)}$, $u|_{op}^{(MAP)} - \Delta u^{(MAP)}$, and $2\Delta u^{(MAP)}$ are written as $\Lambda|_{op}^{(MAP)} f_{\Delta\Lambda}(\Delta \bar{\theta}^{(MAP)})$, $\Lambda|_{op}^{(MAP)} f_{\Delta\Lambda}(-\Delta \bar{\theta}^{(MAP)})$, and $2\Delta \bar{\theta}^{(MAP)}$, respectively. The Jacobian contains the linearized contribution from mooring restoring.

$$D^{(MAP)} = \frac{dY^{(MAP)}}{du} \Big|_{op} = \frac{Y(z|_{op} + \Delta z, u|_{op} + \Delta u, t|_{op}) - Y(z|_{op} - \Delta z, u|_{op} - \Delta u, t|_{op})}{2\Delta u} \Big|_{op}^{(MAP)} \quad (7)$$

While MAP++ and AeroDyn are both modules with constraint states, their numerical linearization approach differs. Equation (7) involves the total derivative of the MAP++ output equations relative to the inputs about the OP, where it is implied that the constraint perturbations, $\Delta z^{(MAP)}$ and $-\Delta z^{(MAP)}$, are tied to the input perturbations, $\Delta u^{(MAP)}$ and $-\Delta u^{(MAP)}$, by solving the implicit constraint equations, $0 = Z(z|_{op} + \Delta z, u|_{op} + \Delta u, t|_{op}) \Big|_{op}^{(MAP)}$ and $0 = Z(z|_{op} - \Delta z, u|_{op} - \Delta u, t|_{op}) \Big|_{op}^{(MAP)}$, respectively, essentially internally eliminating the constraint states within MAP++ during the linearization process. This is in contrast to the approach taken to linearize AeroDyn, whereby the Jacobians $\frac{\partial Z}{\partial z} \Big|_{op}^{(AD)}$, $\frac{\partial Z}{\partial u} \Big|_{op}^{(AD)}$, $\frac{\partial Y}{\partial z} \Big|_{op}^{(AD)}$, and $\frac{\partial Y}{\partial u} \Big|_{op}^{(AD)}$ are computed separately and algebraically manipulated to compute $D^{(AD)}$ as

shown in Eqs. (6) and (7) of [2]. This difference in implementation was needed in MAP++ because the residual of constraint equations, $Z^{(MAP)}$, is not readily available internally within MAP++. The default perturbation sizes are hard-coded within MAP++—but can be customized by recompiling—and are fractions of the water depth for the inputs. To obtain an accurate Jacobian, it is important that the perturbation sizes are not too large to make a taut mooring line go slack or a slack line go taut, but not too small to prevent the tolerance used to solve the implicit constraint equations from effecting the numerically computed derivative.

MODULE-TO-MODULE, INPUT-OUTPUT COUPLING RELATIONSHIPS LINEARIZATION

Most module inputs and outputs in OpenFAST reside on spatial boundaries, which in the FAST modularization framework are defined in terms of a mesh. The module-to-module, input-output coupling relationships in the OpenFAST glue code are algebraic, and include spatial mesh-to-mesh mapping. The general approach to linearizing the mesh mapping within the module-to-module, input-output coupling relationships within OpenFAST, including rotations, is detailed in [2].

The linearized input-output transformation functions, U , are given by Eq. (8) from [2] or Eq. (15) from [4], repeated in Eq. (8) for convenience.

$$0 = \frac{\partial U}{\partial \tilde{u}} \Big|_{op} \Delta u + \frac{\partial U}{\partial y} \Big|_{op} \Delta y \quad \text{with} \quad \left| \frac{\partial U}{\partial \tilde{u}} \Big|_{op} \right| \neq 0 \quad (8)$$

As is evident from Table 1, the InflowWind, ServoDyn, ElastoDyn, AeroDyn, BeamDyn, HydroDyn, and MAP++ modules were developed so that for the most part—other than mapping between independent spatial discretizations—the input of one module equals the output of another. It follows

$$\text{that with } U = \begin{Bmatrix} U^{(Ifw)} \\ U^{(SrvD)} \\ U^{(ED)} \\ U^{(BD)} \\ U^{(AD)} \\ U^{(HD)} \\ U^{(MAP)} \end{Bmatrix}, \quad \Delta u = \begin{Bmatrix} \Delta u^{(Ifw)} \\ \Delta u^{(SrvD)} \\ \Delta u^{(ED)} \\ \Delta u^{(BD)} \\ \Delta u^{(AD)} \\ \Delta u^{(HD)} \\ \Delta u^{(MAP)} \end{Bmatrix}, \quad \text{and}$$

$$\Delta y = \begin{Bmatrix} \Delta y^{(IYW)} \\ \Delta y^{(SrvD)} \\ \Delta y^{(ED)} \\ \Delta y^{(BD)} \\ \Delta y^{(AD)} \\ \Delta y^{(HD)} \\ \Delta y^{(MAP)} \end{Bmatrix}, \text{ the Jacobian matrices evaluated at the OP}$$

from Eq. (8) for these seven modules are given by Eq. (9), where I and 0 are appropriately sized identity and zero matrices and the sub-Jacobian matrices are composed of I s, 0 s, and the linearized matrices from the mapping transfers given in [2].

$$\left. \frac{\partial U}{\partial \tilde{u}} \right|_{op} = \begin{bmatrix} I & 0 & 0 & 0 & \frac{\partial U^{(IYW)}}{\partial \tilde{u}^{(AD)}} & 0 & 0 \\ 0 & I & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & \frac{\partial U^{(ED)}}{\partial \tilde{u}^{(BD)}} & \frac{\partial U^{(ED)}}{\partial \tilde{u}^{(AD)}} & \frac{\partial U^{(ED)}}{\partial \tilde{u}^{(HD)}} & \frac{\partial U^{(ED)}}{\partial \tilde{u}^{(MAP)}} \\ 0 & 0 & 0 & \frac{\partial U^{(BD)}}{\partial \tilde{u}^{(BD)}} & \frac{\partial U^{(BD)}}{\partial \tilde{u}^{(AD)}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\partial U^{(AD)}}{\partial \tilde{u}^{(AD)}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\partial U^{(HD)}}{\partial \tilde{u}^{(HD)}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I \end{bmatrix}_{op} \quad (9a)$$

$$\left. \frac{\partial U}{\partial y} \right|_{op} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial U^{(SrvD)}}{\partial y^{(ED)}} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\partial U^{(ED)}}{\partial y^{(SrvD)}} & \frac{\partial U^{(ED)}}{\partial y^{(ED)}} & \frac{\partial U^{(ED)}}{\partial y^{(BD)}} & \frac{\partial U^{(ED)}}{\partial y^{(AD)}} & \frac{\partial U^{(ED)}}{\partial y^{(HD)}} & \frac{\partial U^{(ED)}}{\partial y^{(MAP)}} \\ 0 & 0 & \frac{\partial U^{(BD)}}{\partial y^{(ED)}} & \frac{\partial U^{(BD)}}{\partial y^{(BD)}} & \frac{\partial U^{(BD)}}{\partial y^{(AD)}} & 0 & 0 \\ \frac{\partial U^{(AD)}}{\partial y^{(IYW)}} & 0 & \frac{\partial U^{(AD)}}{\partial y^{(ED)}} & \frac{\partial U^{(AD)}}{\partial y^{(BD)}} & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial U^{(HD)}}{\partial y^{(ED)}} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial U^{(MAP)}}{\partial y^{(ED)}} & 0 & 0 & 0 & 0 \end{bmatrix}_{op} \quad (9b)$$

Because of their large size, the sub-Jacobian matrices are not shown here, but are described qualitatively instead:

- The first, second, fourth, and fifth equations of Eq. (8) for InflowWind inputs, $\theta = U^{(IYW)}$, ServoDyn inputs, $\theta = U^{(SrvD)}$, BeamDyn inputs, $\theta = U^{(BD)}$, and AeroDyn inputs, $\theta = U^{(AD)}$, are described in [2,3].
- The third equation of Eq. (8) for ElastoDyn inputs, $\theta = U^{(ED)}$, are also described in [2,3], except that the new floating offshore linearization functionality now also includes terms expressing the contributions of HydroDyn and MAP++. That is, this equation

expresses that the applied point force and moment perturbations distributed along the blades and tower as input to ElastoDyn are derived from the aerodynamic-applied line (per-unit length) force and moment perturbations distributed along the blades and tower as output from AeroDyn. This linearized load-mapping transfer also depends on the translational-displacement perturbations of analysis nodes along the blades and tower output from ElastoDyn. And when BeamDyn is enabled, point force and moment perturbations on the hub as input to ElastoDyn are derived from the blade-root reaction point force and moment output from BeamDyn, which also depends on the translational-displacement perturbation of the hub reference point output from ElastoDyn. Additionally, the blade-pitch-angle-commands, nacelle-yaw-moment, and generator-torque perturbations as input to ElastoDyn are derived from the equivalent outputs from ServoDyn. For the floating offshore functionality, point force and moment perturbations on the platform as input to ElastoDyn are derived from the hydrodynamic-applied line (per-unit length) and point force and moment perturbations distributed along the floating platform as output from HydroDyn and the reaction point forces (tensions) lumped at each fairlead as output from MAP++, which also depends on the translational-displacement perturbation of the platform reference point output from ElastoDyn.

- The sixth equation of Eq. (8) for HydroDyn inputs, $\theta = U^{(HD)}$, expresses that the translational displacement, orientation, translational and rotational velocity, and translational and rotational acceleration perturbations of analysis nodes distributed along the floating platform as input to HydroDyn are derived from motion outputs from ElastoDyn at the platform reference point.
- The seventh equation of Eq. (8) for MAP++ inputs, $\theta = U^{(MAP)}$, expresses that the translational displacement perturbations of each fairlead as input to MAP++ are derived from the motion outputs from ElastoDyn at the platform reference point.

The Jacobian, $\left. \frac{\partial U}{\partial \tilde{u}} \right|_{op}$, has ones along its entire diagonal

and it is easily shown that its determinant from Eq. (8) is nonzero.

FINAL MATRIX ASSEMBLY

Once all individual modules and input-output relationships are linearized about the OP, the linearized model of the complete coupled system can be assembled. Linearization of the full-system model produces a linear state-space model representation of the complete nonlinear system about the OP, including the influence of system state and input perturbations

on the system response and outputs. The general linearized form of the complete coupled system is given by Eqs. (18) and

$$(19) \text{ from [4]. With } \Delta x = \left\{ \Delta x^{(ED)^T} \quad \Delta x^{(BD)^T} \quad \Delta x^{(HD)^T} \right\}^T, \text{ the}$$

form for the OpenFAST features linearized to date (without discrete-time states and with ElastoDyn, BeamDyn, and HydroDyn as the only modules with continuous-time states) is given by Eq. (10), where Δu^+ is the additional input perturbations (explained further in [4]).

$$\Delta \dot{x} = A \Delta x + B \Delta u^+ \quad (10a)$$

$$\Delta y = C \Delta x + D \Delta u^+ \quad (10b)$$

The full-system state-space matrices are given in Eq. (11), where $G|_{op}$ —explained more in [4] for the general case—for the OpenFAST linearization to date is given by Eq. (12). The matrix $G|_{op}$ has ones along its entire diagonal and it is easily shown that its determinant from Eq. (12) is nonzero, which means that the matrix inverse, $[G|_{op}]^{-1}$ from Eq. (11), exists and is bounded in the neighborhood around the OP.

The input-transmission matrices impact all matrices of the linearized coupled system, highlighting the important role played by direct feedthrough of input to output in the coupled system response. For example, while the continuous-state

matrix of ElastoDyn, $A^{(ED)}$, contains mass, stiffness, and damping only directly associated with the structural model, the full-system continuous state matrix, A , contains mass, stiffness, and damping associated with coupled aero-hydro-servo-elastics, including hydrodynamic added mass, wave-radiation and viscous damping, and hydrostatic and mooring restoring for FOWTs.

When the linearized full-system matrices A , B , C , and D are exported to a file by OpenFAST, the additional input perturbations, Δu^+ , can be chosen by the user to be: 1) the inputs of all modules, 2) none of the module inputs (removing B and D from the file), or 3) a standard subset of these inputs, which include the standard wind turbine control inputs of nacelle-yaw moment, generator torque, and blade-pitch-angle commands (both independent and rotor-collective); the standard wind-inflow disturbances of horizontal wind speed, power-law shear exponent, and wind-propagation direction; and the standard incident-wave disturbance of wave elevation. Likewise, the output perturbations, Δy , can be chosen by the user to be: 1) the outputs of all modules, 2) none of the module outputs (removing C and D from the file), or 3) only the subset of output variables selected by the user through the OpenFAST module input file(s). Regardless of what the user selects to be exported to a file, all of the module inputs and outputs are used to form the linearized full-system matrices in Eq. (11), but only a subset of these matrices are exported based on the user selection.

$$A = \begin{bmatrix} A^{(ED)} & 0 & 0 \\ 0 & A^{(BD)} & 0 \\ 0 & 0 & A^{(HD)} \end{bmatrix} - \begin{bmatrix} 0 & 0 & B^{(ED)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & B^{(BD)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & B^{(HD)} \end{bmatrix} \left[G|_{op} \right]^{-1} \frac{\partial U}{\partial y} \Big|_{op} \begin{bmatrix} 0 & 0 & 0 \\ C^{(ED)} & 0 & 0 \\ 0 & C^{(BD)} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & C^{(HD)} \\ 0 & 0 & 0 \end{bmatrix} \quad (11a)$$

$$B = \begin{bmatrix} 0 & 0 & B^{(ED)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & B^{(BD)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & B^{(HD)} \end{bmatrix} \left[G|_{op} \right]^{-1} \frac{\partial U}{\partial \bar{u}} \Big|_{op} \quad (11b)$$

$$C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ C^{(ED)} & 0 & 0 \\ 0 & C^{(BD)} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & C^{(HD)} \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} D^{(JW)} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & D^{(SrvD)} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & D^{(ED)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & D^{(BD)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & D^{(AD)} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & D^{(HD)} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & D^{(MAP)} \end{bmatrix} \left[G|_{op} \right]^{-1} \frac{\partial U}{\partial y} \Big|_{op} \begin{bmatrix} 0 & 0 & 0 \\ C^{(ED)} & 0 & 0 \\ 0 & C^{(BD)} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & C^{(HD)} \\ 0 & 0 & 0 \end{bmatrix} \quad (11c)$$

$$D = \begin{bmatrix} D^{(jW)} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & D^{(SrvD)} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & D^{(ED)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & D^{(BD)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & D^{(AD)} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & D^{(HD)} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & D^{(MAP)} \end{bmatrix} \left[G_{lop} \right]^{-1} \frac{\partial U}{\partial \bar{u}} \Big|_{lop} \quad (11d)$$

$$G_{lop} = \begin{bmatrix} I & 0 & 0 & 0 & \frac{\partial U^{(jW)}}{\partial \bar{u}^{(AD)}} & 0 & 0 \\ 0 & I & \frac{\partial U^{(SrvD)}}{\partial y^{(ED)}} D^{(ED)} & 0 & 0 & 0 & 0 \\ 0 & \frac{\partial U^{(ED)}}{\partial y^{(SrvD)}} D^{(SrvD)} & I + \frac{\partial U^{(ED)}}{\partial y^{(ED)}} D^{(ED)} & \frac{\partial U^{(ED)}}{\partial \bar{u}^{(BD)}} + \frac{\partial U^{(ED)}}{\partial y^{(BD)}} D^{(BD)} & \frac{\partial U^{(ED)}}{\partial \bar{u}^{(AD)}} + \frac{\partial U^{(ED)}}{\partial y^{(AD)}} D^{(AD)} & \frac{\partial U^{(ED)}}{\partial \bar{u}^{(HD)}} + \frac{\partial U^{(ED)}}{\partial y^{(HD)}} D^{(HD)} & \frac{\partial U^{(ED)}}{\partial \bar{u}^{(MAP)}} + \frac{\partial U^{(ED)}}{\partial y^{(MAP)}} D^{(MAP)} \\ 0 & 0 & \frac{\partial U^{(BD)}}{\partial y^{(ED)}} D^{(ED)} & \frac{\partial U^{(BD)}}{\partial \bar{u}^{(BD)}} + \frac{\partial U^{(BD)}}{\partial y^{(BD)}} D^{(BD)} & \frac{\partial U^{(BD)}}{\partial \bar{u}^{(AD)}} + \frac{\partial U^{(BD)}}{\partial y^{(AD)}} D^{(AD)} & 0 & 0 \\ \frac{\partial U^{(AD)}}{\partial y^{(jW)}} D^{(jW)} & 0 & \frac{\partial U^{(AD)}}{\partial y^{(ED)}} D^{(ED)} & \frac{\partial U^{(AD)}}{\partial y^{(BD)}} D^{(BD)} & \frac{\partial U^{(AD)}}{\partial \bar{u}^{(AD)}} & 0 & 0 \\ 0 & 0 & \frac{\partial U^{(HD)}}{\partial y^{(ED)}} D^{(ED)} & 0 & 0 & \frac{\partial U^{(HD)}}{\partial \bar{u}^{(HD)}} & 0 \\ 0 & 0 & \frac{\partial U^{(MAP)}}{\partial y^{(ED)}} D^{(ED)} & 0 & 0 & 0 & I \end{bmatrix} \quad \text{with } |G_{lop}| \neq 0 \quad (12)$$

CONCLUSIONS

Linearization of the underlying nonlinear wind-system equations is often important for understanding the system response and exploiting well-established methods and tools for analyzing linear systems. This paper has presented the development of the new linearization functionality of the open-source engineering tool OpenFAST for FOWTs, as well as the concepts and mathematical background needed to understand and apply it. Details have been presented on finding an OP, linearizing the underlying nonlinear equations of each module about the OP, linearizing the module-to-module input-output coupling relationships about the OP, and combining all linearized matrices into the full-system linear state-space model. The new linearization functionality enables the contributions from state-space-based wave excitation, hydrodynamic added mass, state-space-based wave-radiation damping, hydrostatic restoring, linearized viscous drag, and linearized mooring restoring for FOWTs to be included in the full linearized system.

Unfortunately, the implementation at the time of this writing has not yet been completed enough to produce results. Results will be presented in future work to highlight the functionality and verify the implementation (e.g., for a sample case whereby the natural frequencies and damping of the NREL 5-MW baseline wind turbine atop the OC3-Hywind spar buoy will be calculated as a function of rotational speed and presented in Campbell-diagram form).

Beyond the developments implemented here, potential future OpenFAST linearization enhancements include: 1) introducing routines to find a static equilibrium, steady-state, or periodic steady-state condition for improved OP determination; 2) enabling the linearization of the tuned-mass-damper (TMD)

models and external user-specified controllers within ServoDyn for improved controls development and analysis; 3) implementing a form of unsteady airfoil aerodynamics, including dynamic stall, within AeroDyn amendable to linearization (e.g., from [8]) and including that model within the linearization of AeroDyn for advanced aerodynamic and stability analysis; 4) linearizing the offshore functionality of OpenFAST for bottom-fixed offshore wind systems; and 5) linearizing new features as the OpenFAST models are further developed in the future going forward.

ACKNOWLEDGMENTS

The authors would like to thank Bonnie Jonkman of Envision Energy USA, Ltd. for her contributions to the development of the FAST modularization framework and to implementation of the land-based linearization functionality.

This work was authored by the National Renewable Energy Laboratory, operated by the Alliance for Sustainable Energy, LLC, for the U. S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding was provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy. The views expressed in the article do not necessarily represent the views of the U.S. Department of Energy or the U.S. government. The U.S. government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. government purposes.

REFERENCES

- [1] Web page <https://nwtc.nrel.gov/OpenFAST> (accessed May 7, 2018).

- [2] Jonkman, J. M. and Jonkman, B. J. “FAST modularization framework for wind turbine simulation: full-system linearization.” *Journal of Physics: Conference Series, The Science of Making Torque from Wind (TORQUE 2016)*, 5–7 October 2016, Munich, Germany [online journal]. 082010. Vol. 753, 2016. URL: <http://iopscience.iop.org/article/10.1088/1742-6596/753/8/082010/pdf>; NREL/CP-5000-67015. Golden, CO: National Renewable Energy Laboratory.
- [3] Jonkman, J. M.; Jonkman, B. J.; and Platt, A. “Full-System Linearization for Wind Turbines with Aeroelastically Tailored Rotor Blades in OpenFAST.” (in preparation).
- [4] Jonkman, J. M. “The New Modularization Framework for the FAST Wind Turbine CAE Tool.” *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 7–10 January 2013, Grapevine (Dallas/Ft. Worth Region), TX [online proceedings]. URL: <http://arc.aiaa.org/doi/pdf/10.2514/6.2013-202>. AIAA-2013-0202. Reston, VA: American Institute of Aeronautics and Astronautics, January 2013; NREL/CP-5000-57228. Golden, CO: National Renewable Energy Laboratory.
- [5] Duarte, T.; Alves, M.; Jonkman, J.; and Sarmento, A. “State-Space Realization of the Wave-Radiation Force within FAST.” *32nd International Conference on Ocean, Offshore, and Arctic Engineering (OMAE2013)*, 9–14 June 2013, Nantes, France [DVD-ROM]. OMAE2013-10375. Houston, TX: The American Society of Mechanical Engineers (ASME International) Ocean, Offshore and Arctic Engineering (OOAE) Division, June 2013; ISBN 978-0-7918-5542-3; NREL/CP-5000-58099. Golden, CO: National Renewable Energy Laboratory.
- [6] Web page https://nwtc.nrel.gov/SS_Fitting (accessed May 7, 2018).
- [7] Borgman, L. E. “Spectral Analysis of Ocean Wave Forces on Piling.” *Journal of the Waterways and Harbors Division*, Vol. 93, Issue 2, pp. 129-156, 1967.
- [8] Hansen, M. H.; Guanaa, M.; and Madsen, H. A. *A Beddoes-Leishman type dynamic stall model in state-space and indicial formulations*. Risø-R-1354 (EN). Roskilde, Denmark: Risø National Laboratory. 2004.

ANNEX A

LINEAR STATE-SPACE-BASED WAVE-EXCITATION MODEL

In the frequency domain, the potential-flow-based wave-excitation loads (three forces and three moments) can be expressed for unidirectional waves by Eq. (A.1), where $X(\omega, \beta)$ is the frequency- (ω) and direction- (β) dependent

wave-excitation loads on the floating platform normalized per unit wave amplitude (derived, e.g., from a potential-flow-based wave-body interaction solver WAMIT) and $\zeta(\omega)$ is the wave-elevation spectrum for waves propagating in direction β (both are complex-valued to include phases). For time-domain simulations, the spectral wave data and wave direction are model inputs and the phases of the various wave components are selected randomly. The time-domain wave-excitation loads, $F_{Exctn}(t)$, are usually precomputed at model initialization by calculating the inverse Fourier transform shown in Eq. (A.2), where $j = \sqrt{-1}$ is the imaginary number (HydroDyn uses a discrete Fourier transform in place of the continuous-time transform shown).

$$F_{Exctn}(\omega) = X(\omega, \beta) \zeta(\omega) \quad (A.1)$$

$$F_{Exctn}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega, \beta) \zeta(\omega) e^{j\omega t} d\omega \quad (A.2)$$

Equivalently, as shown in [A.1], the wave-excitation load components can be calculated through the infinite convolution integral given by Eq. (3), where the incident wave-excitation kernels derived from Eq. (A.3), represent the incident wave-excitation impulse-response functions (IRF) that depend only on platform parameters. The goal is to use a linear state-space equation to approximate Eq. (3), with the time-domain wave elevation, $\zeta(t)$, acting as the input, and the time-domain wave-excitation loads, $F_{Exctn}(t)$, as the output.

$$K_{Exctn}(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega, \beta) e^{j\omega t} d\omega \quad (A.3)$$

Fitting a linear state-space model to the infinite convolution integral is complicated by the noncausality of this system. This issue has been studied extensively in [A.2] and [A.3], where a time-shifting method is used to arrive at a causal transfer function between wave elevation and wave-excitation loads. We follow the same method used in [A.2] and [A.3] here. Figure A.1 shows an example platform-pitch response (blue curve) resulting from a wave-elevation impulse at time $t = 0$ s. Clearly, the system is noncausal because its response is nonzero for negative time even though the impulse acts at $t = 0$ s. The system response for negative time is as large as the response for positive time, so it cannot be ignored.

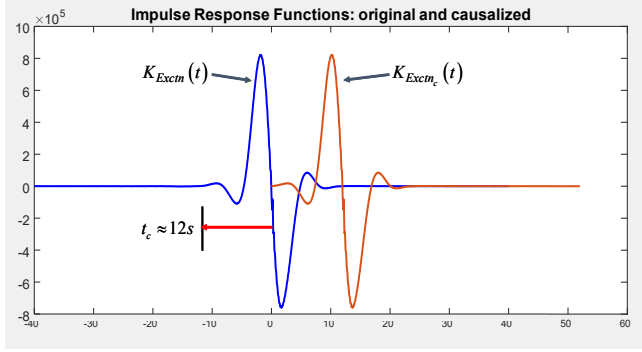


FIGURE A.1. NONCAUSAL AND CAUSAL IMPULSE-RESPONSE FUNCTIONS.

An approximately causal IRF is obtained through a time-shifting method. Figure A.1 shows that for all times less than $-t_c$ ($t_c \approx 12$ s for this example), the IRF is approximately zero. A time delay, t_c , is introduced to shift the original IRF to obtain a “causalised” IRF, $K_{Excmt_c}(t) = K_{Excmt}(t - t_c)$. Figure A.1 shows that this time-shifted IRF (red) is approximately zero for negative time. Equation (A.4), in terms of $K_{Excmt}(t)$ and $\zeta_c(t)$, follows directly from Eq. (3) and the time-shift properties of Fourier transforms, where $\zeta_c(t) = \zeta(t + t_c)$ is the wave elevation predicted at time t_c into the future. To implement this approach, a wave-prediction model must be used to predict the wave elevation for future times. Within OpenFAST simulations, this is not an issue because the wave elevation is precalculated at model initialization, so the future wave elevations are already available. In practice, the future wave evaluation could be estimated (e.g., by measuring the wave elevation with a buoy and approximating the wave evolution).

$$F_{Excmt}(t) = \int_{-\infty}^{\infty} K_{Excmt_c}(t - \tau) \zeta_c(\tau) d\tau \quad (A.4)$$

Now the goal is to fit a linear state-space model to the causalised IRF, $K_{Excmt_c}(t)$, based on input $\zeta_c(t)$. For a state-space system of the form shown in Eq. (4), it can be shown that the IRF is $C_{Excmt} e^{A_{Excmt}t} B_{Excmt}$, where $e^{A_{Excmt}t}$ is the matrix-exponential function [A.4]. The goal is to determine A_{Excmt} , B_{Excmt} , and C_{Excmt} so that $C_{Excmt} e^{A_{Excmt}t} B_{Excmt}$ approximates $K_{Excmt_c}(t)$ in terms of some error norm. This is usually done by minimizing a cost function, such as given by Eq. (A.5), where $G(t_k)$ is a weighting function to be chosen, evaluated at specific sampling times, t_k , for $k = 1, 2, \dots, m$. At least m

discrete values of K_{Excmt_c} are assumed to be known either from theory or experiment.

$$Q = \sum_{k=1}^m G(t_k) \left[K_{Excmt_c}(t_k) - C_{Excmt} e^{A_{Excmt}t_k} B_{Excmt} \right]^2 \quad (A.5)$$

Several methods exist for performing this system-identification step. For this work, we rely on the use of a linear state-space realization of the impulse response using Hankel singular value decomposition and MATLAB’s System-Identification-Toolbox command *impzss* [A.5]. The state-space model structure is selected so that the infinite-frequency limit of the identified model in the frequency domain is zero, forcing the output transmission (D) matrix in the general state-space description to be equal to the zero matrix [A.5]. A balanced model-truncation method is used to arrive at a low-order state-space approximation for the wave-excitation loads.

This general method for generating a linear state-space model of the wave excitation for inclusion into OpenFAST is being developed and programmed into a MATLAB script. Further details, with example cases and a user’s guide for this script will be described in a future publication.

- [A.1] Jonkman, J. M. *Dynamics Modeling and Loads Analysis of an Offshore Floating Wind Turbine*. Ph.D. Thesis. Department of Aerospace Engineering Sciences, University of Colorado, Boulder, CO, 2007; NREL/TP-500-41958. Golden, CO: National Renewable Energy Laboratory.
- [A.2] Yu, Z. and Falnes, J. “State-space Modeling of a Vertical Cylinder in Heave.” *Applied Ocean Research*, Vol. 17, Issue 5, pp. 265-275, 1995.
- [A.3] Lemmer, F., Raach, S., Schlipf, D., and Cheng, P. “Parametric Wave Excitation Model for Floating Wind Turbines.” *Energy Procedia*, Vol. 94, 2016, pp. 290-305; *13th Deep Sea Offshore Wind R&D Conference, EERA DeepWind’2016, 20-22 January 2016, Trondheim, Norway*; DOI: 10.1016/j.egypro.2016.09.186.
- [A.4] Ogata, K. *Modern Control Engineering*, Englewood Cliffs, NJ: Prentice-Hall Inc., 1990.
- [A.5] Ljung, L. and Singh, R. “Version 8 of the MATLAB System Identification Toolbox.” *16th IFAC Symposium on System Identification, Brussels, Belgium*. ISBN 9783902823069; pp. 1826-1832, 2012; doi:10.3182/20120711-3-BE-2027.00061.