

Partitioned nonlinear structural analysis of wind turbines using BeamDyn

Qi Wang^{*1}, Michael A. Sprague^{†1}, and Jason Jonkman^{‡1}

¹National Renewable Energy Laboratory, Golden, CO 80401

In this paper, we present the numerical implementation of BeamDyn, a finite element beam solver based on geometrically exact beam theory (GEBT), and its coupling to different partitions in the FAST modularization framework. The theory is being reviewed first. Then the coupling algorithm and numerical integration scheme specifically designed for wind turbine analysis are introduced. The implicit loose coupling method for BeamDyn coupling is used where nonlinear input-output relations are solved by Newton-Raphson method. Finally, numerical examples are provided to validate both the beam solver and the coupling algorithm.

I. Introduction

Recently, NREL has developed BeamDyn, a nonlinear structural dynamics module for composite wind turbine blade analysis in the FAST modularization framework. This work is motivated by a fact that composite materials have been introduced into the wind energy industry which makes manufacturing of much lighter, larger wind turbine blade a possibility. However, the use of composites also introduces some difficulties in the design and analysis stage. For example, the coupling behavior that exists in composite materials can hardly be captured by the conventional theories which are usually based on isotropic assumptions. Moreover, as the increasing size of the wind turbines, the deformations of the blades cannot be assumed to be linear anymore. This beam solver, BeamDyn, is implemented based on the geometrically exact beam theory (GEBT), which is a beam-deformation model useful in efficient analysis of highly flexible composite structures. GEBT was first proposed by Reissner¹ and then extended to three-dimensional (3D) beams by Simo² and Simo and Vu-Quoc³. Readers are referred to Hodges⁴, in which comprehensive derivations and discussions on nonlinear composite-beam theories can be found. The numerical tool in BeamDyn to discretize the space domain is the Legendre spectral finite elements (LSFE), which is a p -type high-order element that combines the accuracy of global spectral methods with the geometric modeling flexibility of h -type finite elements (FEs). More details on BeamDyn can be found in Wang et al.⁵.

FAST is the flagship multi-physics engineering tool developed at the National Renewable Energy Laboratory (NREL) for analyzing both land-based and offshore wind turbines under realistic operating conditions. The previous beam model in FAST was not capable of predictive analysis of highly flexible, composite wind turbine blades. FAST has been reformulated under a new modularized framework that provides a rigorous means by which various mathematical systems are implemented in distinct modules. The restructuring of FAST greatly enhanced flexibility and expandability to enable further developments of functionality without the need to recode established modules. These modules are interconnected to solve for the globally coupled dynamic responses of wind turbines and wind plants^{6,7}. In previous work, this framework is extended to handle the non-matching spatial grids at interfaces and non-matching temporal meshes that allow module solutions to advance with different time increments and different time integrators.⁸

In this paper, the formulation of the beam theory is firstly reviewed. Then the coupling algorithm of BeamDyn with other modules is presented. Numerical integration scheme that specifically implemented for wind turbine analysis is also introduced. Numerical examples are provided to validate the proposed beam solver and its coupling algorithm when it is running in a couple-to-FAST mode.

II. Geometrically Exact Beam Theory

This section briefly reviews the geometrically exact beam theory. Further details on the content of this section can be found in many other papers and textbooks^{4,9,10}. Figure 1 shows a beam in its initial undeformed and deformed states. A reference frame \mathbf{b}_i is introduced along the beam axis for the undeformed state and a frame \mathbf{B}_i is introduced along each point of the deformed beam axis. The curvilinear coordinate x_1 defines the intrinsic parameterization of the reference line. In this paper, matrix notation is used to denote vectorial or vectorial-like quantities. For example,

^{*}Research Engineer, National Wind Technology Center, AIAA Member. Email: Qi.Wang2@nrel.gov

[†]Senior Research Scientist, Computational Science Center.

[‡]Senior Engineer, National Wind Technology Center, AIAA Professional Member.

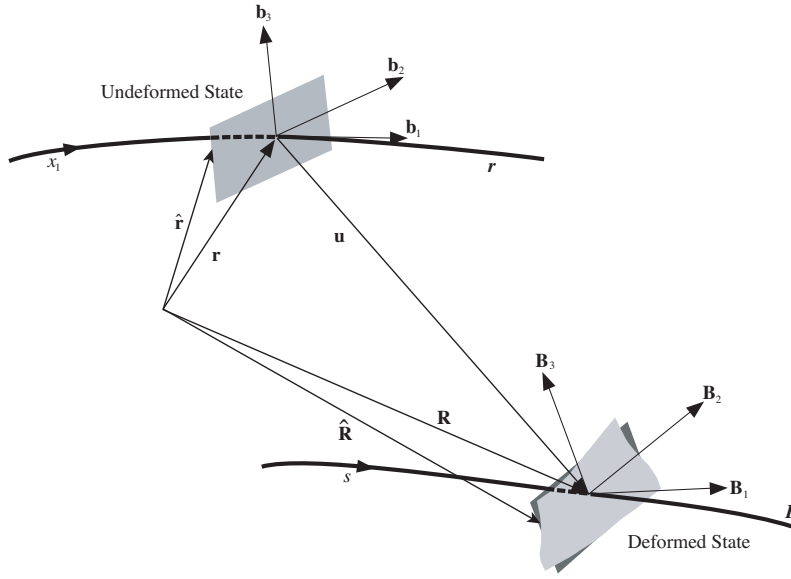


Figure 1: A beam deformation schematic.

an underline denotes a vector \underline{u} , a bar denotes unit vector \bar{n} , and a double underline denotes a tensor $\underline{\underline{A}}$. Note that sometimes the underlines only denote the dimension of the corresponding matrix. The governing equations of motion for geometrically exact beam theory can be written as⁹

$$\dot{\underline{h}} - \underline{F}' = \underline{f} \quad (1)$$

$$\dot{\underline{g}} + \tilde{\underline{u}}\underline{h} - \underline{M}' + (\tilde{\underline{x}}'_0 + \tilde{\underline{u}}')^T \underline{F} = \underline{m} \quad (2)$$

where \underline{h} and \underline{g} are the linear and angular momenta resolved in the inertial coordinate system, respectively; \underline{F} and \underline{M} are the beam's sectional force and moment resultants, respectively; \underline{u} is the one-dimensional (1D) displacement of a point on the reference line; \underline{x}_0 is the position vector of a point along the beam's reference line; and \underline{f} and \underline{m} are the distributed force and moment applied to the beam structure. The notation $(\bullet)'$ indicates a derivative with respect to beam axis x_1 and $\dot{(\bullet)}$ indicates a derivative with respect to time. The tilde operator $\tilde{(\bullet)}$ defines a skew-symmetric tensor corresponding to the given vector. In the literature, it is also termed as “cross-product matrix”. For example,

$$\tilde{\bar{n}} = \begin{bmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{bmatrix}$$

The constitutive equations relate the velocities to the momenta and the 1D strain measures to the sectional resultants as

$$\begin{Bmatrix} \underline{h} \\ \underline{g} \end{Bmatrix} = \underline{\underline{M}} \begin{Bmatrix} \underline{\dot{u}} \\ \underline{\omega} \end{Bmatrix} \quad (3)$$

$$\begin{Bmatrix} \underline{F} \\ \underline{M} \end{Bmatrix} = \underline{\underline{C}} \begin{Bmatrix} \underline{\epsilon} \\ \underline{\kappa} \end{Bmatrix} \quad (4)$$

where $\underline{\underline{M}}$ and $\underline{\underline{C}}$ are the 6×6 sectional mass and stiffness matrices, respectively (note that they are not really tensors); $\underline{\epsilon}$ and $\underline{\kappa}$ are the 1D strains and curvatures, respectively; and, $\underline{\omega}$ is the angular velocity vector that is defined by the rotation tensor \underline{R} as $\underline{\omega} = \text{axial}(\dot{\underline{R}} \underline{R}^T)$. The axial vector \underline{a} associated with a second-order tensor $\underline{\underline{A}}$ is denoted $\underline{a} = \text{axial}(\underline{\underline{A}})$ and its components are defined as

$$\underline{a} = \text{axial}(\underline{\underline{A}}) = \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} = \frac{1}{2} \begin{Bmatrix} A_{32} - A_{23} \\ A_{13} - A_{31} \\ A_{21} - A_{12} \end{Bmatrix} \quad (5)$$

The 1D strain measures are defined as

$$\begin{Bmatrix} \underline{\epsilon} \\ \underline{\kappa} \end{Bmatrix} = \begin{Bmatrix} \underline{x}'_0 + \underline{u}' - (\underline{R} \underline{R}_0) \bar{i}_1 \\ \underline{k} \end{Bmatrix} \quad (6)$$

where $\underline{k} = axial[(\underline{RR}_0)'(\underline{RR}_0)^T]$ is the sectional curvature vector resolved in the inertial basis and \bar{e}_1 is the unit vector along x_1 direction in the inertial basis. Note that these three sets of equations, including equations of motion Eq. (1) and (2), constitutive equations Eq. (3) and (4), and kinematical equations Eq. (6), provide a full mathematical description of elasticity problems.

For a displacement-based finite-element implementation, there are six degrees of freedom at each node: three displacement components and three rotation components. Here, \underline{q} denotes the elemental displacement array as $\underline{q}^T = [\underline{u}^T \ \underline{p}^T]$ where \underline{u} is the displacement and \underline{p} is the rotation-parameter vector. The acceleration array can thus be defined as $\underline{a}^T = [\underline{\ddot{u}}^T \ \underline{\dot{\omega}}^T]$. For nonlinear finite-element analysis, the discretized forms of displacement, velocity, and acceleration are written as

$$\underline{q}(x_1) = \underline{N} \hat{\underline{q}} \quad \underline{q}^T = [\underline{u}^T \ \underline{p}^T] \quad (7)$$

$$\underline{v}(x_1) = \underline{N} \hat{\underline{v}} \quad \underline{v}^T = [\underline{\dot{u}}^T \ \underline{\dot{\omega}}^T] \quad (8)$$

$$\underline{a}(x_1) = \underline{N} \hat{\underline{a}} \quad \underline{a}^T = [\underline{\ddot{u}}^T \ \underline{\dot{\omega}}^T] \quad (9)$$

where \underline{N} is the shape function matrix and (\cdot) denotes a column matrix of nodal values.

III. Implicit coupling algorithm

For clarification, “implicit” here refers to a module that needs information from other modules before the solution can be time-advanced; the modules are thus tied together in a linear or nonlinear system that must be solved for time advancement. Specifically to the BeamDyn module, which is used to represent a blade in the wind turbine system, the inputs are the root motion including displacements/rotations, linear and angular velocities, and linear and angular accelerations; and the outputs are the root reaction forces and moments transferred to the hub. The inputs and outputs can be written as

$$\mathbf{u}_{BD} = [\underline{q} \ \underline{v} \ \underline{a}]^T \quad (10)$$

$$\mathbf{y}_{BD} = [\underline{f} \ \underline{m}]^T \quad (11)$$

The inputs and outputs of the module that being coupled to BeamDyn have the form

$$\mathbf{u}_O = [\underline{f} \ \underline{m}]^T \quad (12)$$

$$\mathbf{y}_O = [\underline{q} \ \underline{v} \ \underline{a}]^T \quad (13)$$

The input-output equations can be written as

$$\mathbf{U}_1 : \mathbf{u}_{BD} - \mathbf{y}_O(\mathbf{u}_O, t) = 0 \quad (14)$$

$$\mathbf{U}_2 : \mathbf{u}_O - \mathbf{y}_{BD}(\mathbf{u}_{BD}, t) = 0 \quad (15)$$

Newton-Raphson method is adopted here to solve this nonlinear system:

$$\begin{bmatrix} \frac{\partial \mathbf{U}_1}{\partial \mathbf{u}_{BD}} & \frac{\partial \mathbf{U}_1}{\partial \mathbf{u}_O} \\ \frac{\partial \mathbf{U}_2}{\partial \mathbf{u}_{BD}} & \frac{\partial \mathbf{U}_2}{\partial \mathbf{u}_O} \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{u}_{BD} \\ \Delta \mathbf{u}_O \end{Bmatrix} = - \begin{Bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \end{Bmatrix} \quad (16)$$

The Jacobian matrix \mathbf{J} on the left-hand-side is computed numerically using the forward differences formulae

$$\mathbf{J}_{ij} = \frac{1}{\epsilon_j} [\mathbf{U}_i(\mathbf{u} + \epsilon_j \mathbf{e}_j) - \mathbf{U}_i(\mathbf{u})] \quad (17)$$

where \mathbf{e}_j is the unit vector in the \mathbf{u}_j direction and ϵ_j represents a small increment.

IV. Trapezoidal Quadrature

The numerical analysis of wind turbine blade features a large number of cross-sectional data along the blade, usually ranging from dozens to more than one hundred stations. A typical practice in dealing with those cross-sectional data is interpolating a subset of the data to the quadrature point, for example, Gauss point. However, there are two drawbacks of this method. One is that it is very difficult to include all the data given the large number of cross sections used in wind turbine modeling. The widely used NREL 5-MW reference wind turbine blade, for example, provides 49 stations along the blade axis. If linearly discretizing this blade, 48 first-order finite elements, with two end points align with two adjacent stations, are needed to make sure that all the cross-sectional data being used in the simulation. The other drawback is that there is no rigorous way to interpolate the station data to the quadrature points.

Given the discussion above, we implemented a new trapezoidal quadrature into BeamDyn, see Eq. 18.

$$\int_a^b f(x)dx \approx \frac{1}{2} \sum_{k=1}^N (x_{k+1} - x_k) [f(x_{k+1}) + f(x_k)] \quad (18)$$

where N is the total number of intervals. Generally, this is a over-integration scheme by assuming that each station point is a quadrature point regardless of the number of the finite element nodes. Based on this assumption, all the cross-sectional data are included in the calculation and no interpolation is needed since the quadrature points are exactly located on the cross-sectional station points.

V. Content of full paper

In the full paper, we will provide more details on the coupling algorithm and the implementation. It is noted that the case discussed in the previous section is only one example among the actual coupled cases in FAST. For example, BeamDyn outputs motions along the beam for use by the aerodynamics module AeroDyn and also receives as input loads along blade from AeroDyn. Numerical examples, including studying of accuracy and stability of the coupling algorithm and verification and validations of the realistic wind turbine blades will be provided.

Acknowledgments

This work was supported by the U.S. Department of Energy under Contract No. DE-AC36-08GO28308 with the National Renewable Energy Laboratory. Funding for the work was provided by the DOE Office of Energy Efficiency and Renewable Energy, Wind and Water Power Technologies Office.

References

- ¹Reissner, E., "On one-dimensional large-displacement finite-strain beam theory," *Studies in Applied Mathematics LII*, 1973, pp. 87–95.
- ²Simo, J. C., "A finite strain beam formulation. The three-dimensional dynamic problem. Part I," *Computer Methods in Applied Mechanics and Engineering*, Vol. 49, 1985, pp. 55–70.
- ³Simo, J. C. and Vu-Quoc, L., "A three-dimensional finite-strain rod model. Part II," *Computer Methods in Applied Mechanics and Engineering*, Vol. 58, 1986, pp. 79–116.
- ⁴Hodges, D. H., *Nonlinear Composite Beam Theory*, AIAA, 2006.
- ⁵Wang, Q., Sprague, M. A., Jonkman, J., and Johnson, N., "Nonlinear Legendre spectral finite elements for wind turbine blade dynamics," *Proceedings of the 32nd ASME Wind Energy Symposium*, National Harbor, Maryland, January 2014.
- ⁶Jonkman, J. M., "The new modularization framework for the FAST wind turbine CAE tool," *Proceedings of the 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Grapevine, Texas, January 2013.
- ⁷Jonkman, J. and Jonkman, B., "FAST v8," <https://nwtc.nrel.gov/FAST8>, October 2013, [Online; accessed 29-OCTOBER-2014].
- ⁸Sprague, M., Jonkman, J., and Jonkman, B., "FAST modular wind turbine CAE tool: non matching spatial and temporal meshes," *Proceedings of the 32nd ASME Wind Energy Symposium*, National Harbor, Maryland, January 2014.
- ⁹Bauchau, O. A., *Flexible Multibody Dynamics*, Springer, 2010.
- ¹⁰Yu, W. and Blair, M., "GEBT: A general-purpose nonlinear analysis tool for composite beams," *Composite Structures*, Vol. 94, 2012, pp. 2677–2689.