The Journal of Open Source Software

GXBeam: A Pure Julia Implementation of Geometrically Exact Beam Theory

Taylor McDonnell[1] **and Andrew Ning**[1]

1 Department of Mechanical Engineering, Brigham Young University, Provo, UT, 84602, USA

DOI: 10.21105/joss.03924

Software

- Review
- Repository
- Archive

Editor: Pending Editor

Submitted: 16 November 2021
Published: 19 November 2021

Summary

When the cross sections of a three-dimensional structure are small compared to the length of the structure, beam theory may be used to efficiently model the structure's three-dimensional behavior. Applications of beam theory include, but are not limited to, the structural modeling of buildings, bridges, aircraft, helicopter blades, and wind turbines. When deflections are small, linear beam theories may be used to model the behavior of slender structures. When deflections become significant, such as encountered when modeling high aspect ratio wings and/or large wind turbine blades, nonlinearities associated with geometric deformations must be accounted for.

Geometrically exact beam theory, as pioneered by Reissner (Reissner, 1973) and extended by Hodges (Hodges, 2006b), captures all of the nonlinearities associated with large deflections and/or rotations, assuming strains are small. It also allows for the coupling terms associated with the use of composite materials to be explicitly modeled as part of the beam analysis. When coupled with a cross sectional analysis, such as a variational asymptotic beam sectional analysis (Hodges, 2006a), it constitutes an accurate and efficient replacement for a full three-dimensional structural analysis.

`GXBeam` is a geometrically exact beam theory package which is written completely in the Julia programming language (Bezanson et al., 2017). It was originally based on the open source code `GEBT` and its associated papers (Wang & Yu, 2017; Yu & Blair, 2012), which adopt the mixed formulation of geometrically exact beam theory presented by Hodges. It has since incorporated several improvements relative to the `GEBT` code. One such improvement is the introduction of a custom rotation parameterization, which is based upon Wiener-Milenkovic parameters, but is singularity free. This improvement increases the codes numerical stability for large deformations and allows for beam deformations in excess of 360 degrees. Additional improvements include the explicit modeling of the influence of point masses and/or rigid bodies, gravitational forces, and forces due to body frame linear/angular accelerations.

Statement of Need

One of the key advantages of `GXBeam` relative to other geometrically exact beam theory codes is that is written completely in the Julia programming language. This presents several advantages for the `GXBeam` package. First, since Julia is a higher-level language, the code is generally easier to develop, maintain, and extend than lower-level languages. This is especially helpful from a research perspective if one wishes to include `GXBeam` as a component of a multidisciplinary design optimization framework and/or fluid structure interaction solver. Second, by leveraging Julia's type system, Julia-specific automatic differentiation packages (such as ForwardDiff (Revels et al., 2016)) may be used to obtain exact derivatives for sensitivity analyses and/or gradient-based optimization. Third, by maintaining type stability and minimizing allocations, this package is able to perform analyses with minimal overhead compared to lower-level languages such as C or Fortran. Finally, the code is able to access and use several well-developed Julia-specific packages to enhance its capabilities such as NLsolve (Mogensen et al., 2020) for solving nonlinear sets of equations, WriteVTK (Polanco, 2021) for writing visualization files, and DifferentialEquations (Rackauckas & Nie, 2017) for solving differential equations.

Even if one were to disregard the advantages associated with the use of the Julia language, `GXBeam` is still one of the most feature-rich open-source geometrically exact beam theory programs available. Rather than restricting analyses to a single beam, `GXBeam` is able to model complex systems of interconnected nonlinear composite beams. `GXBeam` also allows for a wide variety of analyses to be performed including linear and/or nonlinear static, steady state, eigenvalue, and/or time marching analyses. Forces/moments in `GXBeam` may be applied to nodes and/or elements and expressed as arbitrary functions of time. Native support for gravitational loads and reference frame linear/angular velocity and/or acceleration are also supported. Additionally, `GXBeam` allows point masses and/or rigid bodies with potentially time-varying inertial properties to be placed at arbitrary locations throughout each beam assembly.

`GXBeam` may be used as either a standalone tool, or as a component of a larger analysis framework. Its results are designed to be smooth and continuous, so that the package may be used as part of a gradient-based design optimization framework. The package is also designed to be modular, so that it can be used as part of a fluid-structure interaction framework. It has also been verified and validated using analytical, computational, and experimental results so that users may be reasonably confident that the results predicted by GXBeam are correct (when used within the theoretical limitations of geometrically exact beam theory). These verifications and validations are included as part of the package's documentation so the accuracy of the package can be verified by anyone wishing to use it. These features make it an invaluable tool for modeling beams in a geometrically exact manner.

References

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, *59*(1), 65–98. https://doi.org/10.1137/141000671

Hodges, D. H. (2006a). Cross-Sectional Analysis for Beams. In *Nonlinear Composite Beam Theory* (pp. 59–102). American Institute of Aeronautics; Astronautics. https://doi.org/10.2514/5.9781600866821.0059.0102

Hodges, D. H. (2006b). One-Dimensional Theory of Beams. In *Nonlinear Composite Beam Theory* (pp. 103–142). American Institute of Aeronautics; Astronautics. https://doi.org/10.2514/5.9781600866821.0103.0142

Mogensen, P. K., Carlsson, K., Villemot, S., Lyon, S., Gomez, M., Rackauckas, C., Holy, T., Widmann, D., Kelman, T., & Karrasch, D. (2020). *NLsolve.jl*. Zenodo. https://doi.org/10.5281/zenodo.4404703

Polanco, J. I. (2021). *WriteVTK.jl: a Julia package for writing VTK XML files*. Zenodo. https://doi.org/10.5281/zenodo.5633646

Rackauckas, C., & Nie, Q. (2017). DifferentialEquations.jl – A Performant and Feature-Rich Ecosystem for Solving Differential Equations in Julia. *The Journal of Open Research Software*, *5*(1). https://doi.org/10.5334/jors.151

Reissner, E. (1973). On One-Dimensional Large-Displacement Finite-Strain Beam Theory. *Studies in Applied Mathematics*, *52*(2), 87–95. https://doi.org/10.1002/sapm197352287

Revels, J., Lubin, M., & Papamarkou, T. (2016). Forward-mode automatic differentiation in Julia. *arXiv:1607.07892 [cs.MS]*. https://arxiv.org/abs/1607.07892

Wang, Q., & Yu, W. (2017). Geometrically nonlinear analysis of composite beams using Wiener-Milenković parameters. *Journal of Renewable and Sustainable Energy*, *9*(3), 033306. https://doi.org/10.1063/1.4985091

Yu, W., & Blair, M. (2012). GEBT: A general-purpose nonlinear analysis tool for composite beams. *Composite Structures*, *94*(9), 2677–2689. https://doi.org/10.1016/j.compstruct.2012.04.007