The Journal of Open Source Software

pyBEMT: An implementation of the Blade Element Momentum Theory in Python

Knut Erik T. Giljarhus[1]

1 Department of Mechanical and Structural Engineering and Materials Science, University of Stavanger, Stavanger, Norway

DOI: 10.21105/joss.02480

Software

- Review
- Repository
- Archive

Editor: Stefan Pfenninger

Reviewers:

- @rafmudaf
- @antviro

Submitted: 08 June 2020
Published: 13 July 2020

Summary

The use of rotating blades to generate thrust in the form of propellers or torque in the form of turbines is of great significance for transportation and energy generation. This has led to extensive research and development of mathematical models to better understand, predict and optimize the performance of these machines. The blade element momentum theory (BEMT) is one such method with a long history dating back to Glauert (1935).

Despite the development of more sophisticated methods, such as vortex methods(Gohard, 1978), the blade element momentum theory is still considered relevant for the study of rotor design. Its simple formulation lends itself to use in education and to quickly analyse new ideas. As examples of recent uses of BEMT, it was used by Herniczek, Jee, Sanders, & Feszty (2019) to compare airfoils and blade shapes for an unmanned rotorcraft and by Borg, Xiao, Allsop, Incecik, & Peyrard (2020), together with computational fluid dynamics, to analyze the performance of a high-solidity tidal turbine design. Several open-source packages include an implementation of BEMT, for instance the `AeroDyn`(Moriarty & Hansen, 2005) solver used in the whole-turbine simulation software `OpenFAST`("OpenFAST," 2020), and the `QBlade`(Marten, Wendler, Pechlivanoglou, Nayeri, & Paschereit, 2013) software for wind turbine blade design. `QBlade` has later been forked to develop `JBLADE`(Silvestre, Morgado, & Pascoa, 2013), a software focusing on propeller design.

`pyBEMT` is unique in that it offers a unified implementation of the blade element momentum theory, supporting both propellers and turbines. This is of particular interest in education, to demonstrate the similarities between these two domains. The software is designed as a stand-alone Python implementation with emphasis on readability and extensibility. Its modular design and permissive license also makes it suitable for integration into other simulation tools. Other notable features of the package are a model for coaxial rotors and optimization of rotor parameters using the differential evolution algorithm in `SciPy`. The use of coaxial rotor systems has recently seen renewed interest with the growth in unmanned aerial vehicles and autonomous systems.

Figure Figure 1 shows two examples of predictions from the model for a tidal stream turbine(Bahaj, Molland, Chaplin, & Batten, 2007) and an airplane propeller(Theodorsen, Stickle, & Brevoort, 1937), respectively. `pyBEMT` is currently applied in research projects on rotor design for unmanned aerial vehicles and turbine design for tidal stream turbines, as well as used in education within fluid dynamics and computational engineering.

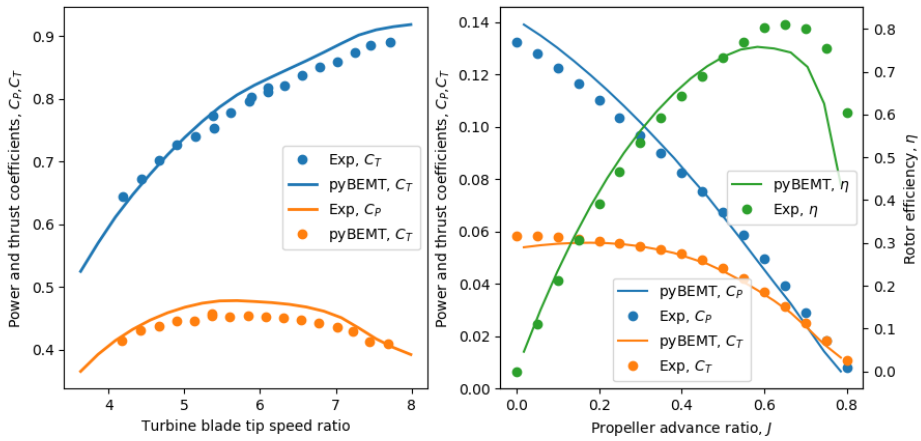

Figure 1: `pyBEMT` applied to a tidal stream turbine (left) and an airplane propeller (right).

Acknowledgements

Jørgen Apeland, Vetle B. Ingebretsen and Stian R. Hidle from the University of Stavanger are acknowledged for their contribution of experimental validation data.

References

Bahaj, A. S., Molland, A. F., Chaplin, J. R., & Batten, W. M. J. (2007). Power and thrust measurements of marine current turbines under various hydrodynamic flow conditions in a cavitation tunnel and a towing tank. *Renewable Energy*, *32*(3), 407–426. doi:10.1016/j.renene.2006.01.012

Borg, M. G., Xiao, Q., Allsop, S., Incecik, A., & Peyrard, C. (2020). A numerical performance analysis of a ducted, high-solidity tidal turbine. *Renewable Energy*. doi:10.1016/j.renene.2020.04.005

Glauert, H. (1935). Airplane propellers. In *Aerodynamic theory* (pp. 169–360). Berlin, Heidelberg: Springer Berlin Heidelberg.

Gohard, J. C. (1978). *Free wake analysis of a wind turbine aerodynamics. Wind energy conversion. ASRL-tr-184-14*. Massachusetts Institute of Technology, Department of Aeronautics; Astronautics, Aeroelastic; Structures Research Laboratory: Cambridge, MA. doi:10.2172/5428795

Herniczek, M. K., Jee, D., Sanders, B., & Feszty, D. (2019). Rotor blade optimization and flight testing of a small uav rotorcraft. *Journal of Unmanned Vehicle Systems*, *7*(4), 325–344. doi:10.1139/juvs-2017-0005

Marten, D., Wendler, J., Pechlivanoglou, G., Nayeri, C. N., & Paschereit, C. O. (2013). QBLADE: An open source tool for design and simulation of horizontal and vertical axis wind turbines. *IJETAE*, *3*(3), 264–269.

Moriarty, P. J., & Hansen, A. C. (2005). *AeroDyn theory manual*. National Renewable Energy Lab., Golden, CO (US). doi:10.2172/15014831

OpenFAST. (2020).*GitHub repository*. GitHub. Retrieved from https://github.com/OpenFAST/openfast

Silvestre, M. A., Morgado, J. P., & Pascoa, J. (2013). JBLADE: A propeller design and analysis code. In *2013 international powered lift conference* (p. 4220). doi:10.2514/6.2013-4220

Theodorsen, T., Stickle, G. W., & Brevoort, M. J. (1937). Characteristics of six propellers including the high-speed range. *Annual Report-National Advisory Committee for Aeronautics*, 401.