The Journal of Open Source Software

Pybotics: Python Toolbox for Robotics

Nicholas A. Nadeau[1]

1 Department of Automated Manufacturing Engineering, École de Technologie Supérieure, Montréal, QC H3C 1K3, Canada

DOI: 10.21105/joss.01738

Software

- Review
- Repository
- Archive

Submitted: 13 September 2019
Published: 30 September 2019

Summary

Modern robotic programming relies on offline simulation to reduce process downtime. In a virtual environment, application specialists can program, visualize, and test their robotic application before deploying it to the real production environment. This offline process saves both time and costs while increasing the safety and efficacy of the robotic application. However, to achieve a high level of fidelity between virtual and production environments, the robot system must be accurate. According to ISO 5725-1, accuracy refers to closeness or *trueness* of measurements to a specific value, while precision refers to the closeness of the measurements to each other. Most industrial robots are inherently precise (i.e., repeatable), but they are not necessarily accurate. One cost-effective approach to obtaining a more accurate robot is through calibration, where the actual kinematic and non-kinematic parameters of the robot model are identified and improved upon when compared to the nominal model.

`Pybotics` is an open-source Python toolbox for robot kinematics and calibration. It was designed to provide a simple, clear, and concise interface to quickly simulate and evaluate common robot concepts, such as kinematics, dynamics, trajectory generations, and calibration. The toolbox is specifically designed for use with the Modified Denavit–Hartenberg parameters convention, introduced and popularized by Craig (2005), which uses four geometric parameters to define reference frames along the links of a robot manipulator.

As MATLAB® is not necessarily readily available outside of academia, `Pybotics` was originally developed as a fully open-source alternative to the `Robotics Toolbox` by Peter Corke (Corke, 2017) with the intention of being used in both research and industry. The `Pybotics` toolbox leverages the `NumPy` package for computational efficiency (Van Der Walt, Colbert, & Varoquaux, 2011) and offers a flexible interface to model robot manipulators using array-based notation. The modelling approach allows for the vectorization of the robot model and integration with the robust optimization algorithms contained in the `SciPy` package (Jones, Oliphant, Peterson, & others, 2001). This results in the capability to easily calibrate a robot model and forms the foundation of the research presented in Nadeau, Bonev, & Joubair (2019). Furthermore, real-time robot optimization applications, an example of which is presented in Nadeau & Bonev (2018), can be augmented with machine learning through the vectorial interface of `Pybotics` and the `Scikit-learn` framework (Pedregosa et al., 2011).

Acknowledgements

We thank Professor Ilian Bonev for his supervision throughout the graduate studies during which this package was developed.

References

Corke, P. (2017). *Robotics, Vision & Control* (Vol. 118). Springer. doi:10.1007/978-3-319-54413-7

Craig, J. J. (2005). *Introduction to Robotics Mechanics and Control* (3rd ed.). Pearson Education International.

Jones, E., Oliphant, T., Peterson, P., & others. (2001). SciPy: Open source scientific tools for Python. Retrieved from http://www.scipy.org/

Nadeau, N. A., & Bonev, I. A. (2018). Evolutionary motion control optimization in physical human-robot interaction. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 1347–1353). IEEE. doi:10.1109/IROS.2018.8593598

Nadeau, N. A., Bonev, I. A., & Joubair, A. (2019). Impedance control self-calibration of a collaborative robot using kinematic coupling. *Robotics*, *8*(2), 33. doi:10.3390/robotics8020033

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., et al. (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research*, *12*(Oct), 2825–2830.

Van Der Walt, S., Colbert, S. C., & Varoquaux, G. (2011). The numpy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, *13*(2), 22. doi:10.1109/MCSE.2011.37