The Journal of Open Source Software

IWOPY: Fraunhofer IWES optimization tools in Python

Jonas Schmidt [1¶] **and Bernhard Stoevesandt** [1]

1 Fraunhofer IWES, Küpkersweg 70, 26129 Oldenburg, Germany ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Frauke Wiese

Reviewers:

- @BenWinchester
- @StewMH

Submitted: 27 March 2023
Published: unpublished

Summary

Optimization problems are described by optimization variables, which are scalars that are modified during the process of optimization; objective functions, which depend on the variables and whose values define the metric for rating the choices of the latter; and constraints, which are also functions of the optimization variables and define validity of the solution. The variables can be discrete or continuous, bounded or unbounded; the number of objectives can be one or many; and constraints can require equality or inequality. Many Python packages formulate a framework for the description of such problems, accompanied by a library of optimizers. Hence, switching from one to another optimization package can often be tedious and a meta-solution is required that can serve as a single interface to multiple optimization packages.

Statement of need

The Python package `iwopy` provides a general object-oriented formulation for optimization problems, objective functions, and optimization constraints. The optimization problem class defines the optimization variables, their bounds and their types. Objectives and constraints can then be added in an independent step to the problem, such that they can easily be exchanged and modified by the user. The framework is general enough for supporting complex science and engineering problems, and it supports single, multi and many objective optimization problems.

The core functionality of `iwopy` is to provide interfaces to other existing Python optimization packages, like `pymoo` (Blank & Deb, 2020) or `pygmo` (Biscani & Izzo, 2020). Once the problem is formulated within the framework sketched above, the optimizers from the supported liked packages can be selected and switched easily.

The design of `iwopy` has a focus on vectorized evaluation approaches, as for example often provided by heuristic algorithms that rely on the concept of populations. If the vectorized evaluation of a complete population of individual choices of optimization variables is implemented by the user, this enables a vast speed-up of optimizations compared to the one-by-one evaluation through a loop.

Acknowledgements

The development of `iwopy` and has been supported through multiple publicly funded research projects. We acknowledge in particular the funding by the Federal Ministry of Economic Affairs and Climate Action (BMWK) through the projects Smart Wind Farms (grant no. 0325851B) and GW-Wakes (0325397B) as well as the funding by the Federal Ministry of Education and Research (BMBF) in the framework of the project H2Digital (03SF0635).

References

Biscani, F., & Izzo, D. (2020). A parallel global multiobjective framework for optimization: pagmo. *Journal of Open Source Software*, *5*(53), 2338. https://doi.org/10.21105/joss.02338

Blank, J., & Deb, K. (2020). Pymoo: Multi-objective optimization in python. *IEEE Access*, *8*, 89497–89509.