The Journal of Open Source Software

topoptlab: An Open and Modular Framework for Benchmarking and Research in Topology Optimization

Stefan Hiemer [1,2]

1 Center for Complexity and Biosystems, Department of Physics "Aldo Pontremoli", University of Milan, Via Celoria 16, 20133 Milano, Italy **2** CNR – Consiglio Nazionale delle Ricerche, Istituto di Chimica della Materia Condensata e di Tecnologie per l'Energia, Via R. Cozzi 53, 20125 Milano, Italy

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: George K. Thiruvathukal

Reviewers:

- @mscroggs
- @tisaac

Submitted: 15 September 2025
Published: unpublished

Summary

Topology optimization (TO) is becoming increasingly popular across physics, engineering, and materials science, as it provides a systematic way to discover efficient designs. Given a set of bounded design variables $x_{\min} \leq x_e \leq x_{\max}$ and a cost or objective function $C(x)$, subject to a discretized physical problem $K(x)u = f$ as well as a set of m equality and n inequality constraints, the general TO problem can be formulated as

$$
\begin{aligned}
&\min_{x} \; C(x) \\
&\text{subject to:} \\
&\quad K(x)u = f, \\
&\quad g_i(x) = 0, \quad i = 1, \dots, m, \\
&\quad h_j(x) \leq 0, \quad j = 1, \dots, n, \\
&\quad x_{\min} \leq x_e \leq x_{\max}, \quad \forall e \in \text{elements}.
\end{aligned}
$$

The common TO choice of design representation is density-based material interpolation scheme, where the abstract design variables x_e become relative element-wise densities which scale the physical properties A of each element via simple relationships as in the popular modified SIMP approach (Sigmund, 2007):

$$A(x_e) = A_{min} + (A_0 - A_{min})x_e^k \text{ with } 0 \leq x_e \leq 1$$

where k is a penalization factor ensuring densities close to 0/1, A_0 the property of the full material and A_{min} a small value to prevent singularities in the physical problem. The final design then emerges from the optimal density distribution.

Topoptlab is a modular and transparent framework for research and benchmarking in topology optimization with a focus on clarity, reproducibility, and accessibility as a tool for both research and advanced education.

Statement of need

In TO, it has become longstanding practice to demonstrate new methods with short Matlab scripts(Andreassen et al., 2011; Ferrari & Sigmund, 2020; Sigmund, 2001; Wang et al., 2021). While these codes have played an important role in the spread and development of ideas, they also come with notable limitations: First, Matlab requires a commercial license, and is only partially compatible to its free alternative Octave. Second, extension and combination of state-of-the-art methods demands combining multiple monolithic scripts, some of which are outdated or mutually incompatible. Third, while modern finite element frameworks such as **FEniCS**(Alnæs et al., 2014; Baratta et al., 2023; Scroggs, Dokken, et al., 2022; Scroggs, Baratta, et al., 2022), **deal.II** (Arndt et al., 2021), and **ElmerFEM**(Malinen & Råback, 2013) provide powerful high-performance environments, their abstraction layers tend to complicate access to low-level implementation details which is necessary for research in TO. Also common use cases in TO allow shortcuts such as regular meshes as ideally the geometry emerges during the optimization process or partial negligence of close to empty elements as preconditioning.

Topoptlab was developed to address these challenges by providing a stable and extensible environment tailored to the needs of the TO community. It serves as a library for writing complete problems from scratch in spirit of the already conventional Matlab scripts, and offers a offers a high-level driver routine (topology_optimization) in which users can exchange components (filter, objective function, etc.) by passing custom callables or objects as arguments. It may also serve as a reference implementation which can be used as test case for existing HPC codes that want to incorporate TO in their software.

Topoptlab offers the components needed for TO such as different material interpolation schemes (SIMP, RAMP, and bound-based interpolation), filters for regularization (Bruns & Tortorelli, 2001; Sigmund, 1997), projections Sigmund (2007) and manufacturability (Langelaar, 2017), and finite element implementations for different physical problems (linear elasticity, heat conduction, etc.) with both standard numerical integration and analytically integrated elements generated through *Symfem*(Scroggs, 2021). Constrained optimization is supported through the Method of Moving Asymptotes (MMA) (Svanberg, 1987), the Globally Convergent Method of Moving Asymptotes (GCMMA) (Svanberg, 2002) as inmplemented in (Deetman, 2025) as well as the Optimality Criteria (Andreassen et al., 2011; Bendsoe & Sigmund, 2003), while the solution of the system of equations is done via routines offered by *scipy* (Virtanen et al., 2020), *cvxopt* (Andersen et al., 2020), and also custom implementations of preconditioners like algebraic multigrid or block-preconditioners. Finally, *TopOptLab* offers a number of introductory articles with comments on implementation and contains monolithic scripts that serve as teaching tools as well as an archive for Python translations of important Matlab teaching codes (e. g. [Andreassen et al. (2011); andreassen2014determine]).

Acknowledgments

We acknowledge the support by the Humboldt foundation through the Feodor-Lynen fellowship. We acknowledge partial support from the ARCHIBIOFOAM project which received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101161052. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Innovation Council and SMEs Executive Agency (EISMEA). Neither the European Union nor the granting authority can be held responsible for them.

Special thanks are extended to Christof Schulze for his valuable suggestions on code improvements, to Stefano Zapperi for hosting at the University of Milan and for his continuous support, and to Peter Råback for sharing his vast knowledge on FEM related implementation details.

References

Alnæs, M. S., Logg, A., Ølgaard, K. B., Rognes, M. E., & Wells, G. N. (2014). Unified form language: A domain-specific language for weak formulations of partial differential equations. *ACM Transactions on Mathematical Software (TOMS)*, *40*(2), 1–37. https://doi.org/10.1145/2566630

Andersen, M., Dahl, J., & Vandenberghe, L. (2020). CVXOPT: Convex optimization. *Astrophysics Source Code Library*, ascl–2008. https://cvxopt.org

Andreassen, E., Clausen, A., Schevenels, M., Lazarov, B. S., & Sigmund, O. (2011). Efficient topology optimization in MATLAB using 88 lines of code. *Structural and Multidisciplinary Optimization*, *43*(1), 1–16. https://doi.org/10.1007/s00158-010-0594-7

Arndt, D., Bangerth, W., Davydov, D., Heister, T., Heltai, L., Kronbichler, M., Maier, M., Pelteret, J.-P., Turcksin, B., & Wells, D. (2021). The deal. II finite element library: Design, features, and insights. *Computers & Mathematics with Applications*, *81*, 407–422. https://doi.org/10.1016/j.camwa.2020.02.022

Baratta, I. A., Dean, J. P., Dokken, J. S., Habera, M., HALE, J., Richardson, C. N., Rognes, M. E., Scroggs, M. W., Sime, N., & Wells, G. N. (2023). *DOLFINx: The next generation FEniCS problem solving environment*. https://doi.org/10.5281/zenodo.10447666

Bendsoe, M. P., & Sigmund, O. (2003). *Topology optimization: Theory, methods, and applications*. Springer Science & Business Media. https://doi.org/10.1007/978-3-662-05086-6

Bruns, T. E., & Tortorelli, D. A. (2001). Topology optimization of non-linear elastic structures and compliant mechanisms. *Computer Methods in Applied Mechanics and Engineering*, *190*(26-27), 3443–3459. https://doi.org/10.1016/S0045-7825(00)00278-4

Deetman, A. (2025). *GCMMA-MMA-python: Python implementation of the method of moving asymptotes* (Version 0.3.1). Zenodo. https://doi.org/10.5281/zenodo.15459165

Ferrari, F., & Sigmund, O. (2020). A new generation 99 line matlab code for compliance topology optimization and its extension to 3D. *Structural and Multidisciplinary Optimization*, *62*(4), 2211–2228. https://doi.org/10.1007/s00158-020-02629-w

Guest, J. K., Prévost, J. H., & Belytschko, T. (2004). Achieving minimum length scale in topology optimization using nodal design variables and projection functions. *International Journal for Numerical Methods in Engineering*, *61*(2), 238–254. https://doi.org/10.1002/nme.1064

Langelaar, M. (2017). An additive manufacturing filter for topology optimization of print-ready designs. *Structural and Multidisciplinary Optimization*, *55*, 871–883. https://doi.org/10.1007/s00158-016-1522-2

Malinen, M., & Råback, P. (2013). Elmer finite element solver for multiphysics and multiscale problems. *Multiscale Modelling Methods for Applications in Material Science*, *19*, 101–113.

Scroggs, M. W. (2021). Symfem: A symbolic finite element definition library. *Journal of Open Source Software*, *6*(64), 3556. https://doi.org/10.21105/joss.03556

Scroggs, M. W., Baratta, I. A., Richardson, C. N., & Wells, G. N. (2022). Basix: A runtime finite element basis evaluation library. *Journal of Open Source Software*, *7*(73), 3982. https://doi.org/10.21105/joss.03982

Scroggs, M. W., Dokken, J. S., Richardson, C. N., & Wells, G. N. (2022). Construction of arbitrary order finite element degree-of-freedom maps on polygonal and polyhedral cell meshes. *ACM Transactions on Mathematical Software (TOMS)*, *48*(2), 1–23. https://doi.org/10.1145/3524456

Sigmund, O. (1997). On the design of compliant mechanisms using topology optimization. *Journal of Structural Mechanics*, *25*(4), 493–524. https://doi.org/10.1080/08905459708945415

Sigmund, O. (2001). A 99 line topology optimization code written in matlab. *Structural and Multidisciplinary Optimization*, *21*(2), 120–127. https://doi.org/10.1007/s001580050176

Sigmund, O. (2007). Morphology-based black and white filters for topology optimization. *Structural and Multidisciplinary Optimization*, *33*, 401–424. https://doi.org/10.1007/s00158-006-0087-x

Svanberg, K. (1987). The method of moving asymptotes—a new method for structural optimization. *International Journal for Numerical Methods in Engineering*, *24*(2), 359–373. https://doi.org/10.1002/nme.1620240207

Svanberg, K. (2002). A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM Journal on Optimization*, *12*(2), 555–573. https://doi.org/10.1137/S1052623499362822

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., … SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, *17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2

Wang, C., Zhao, Z., Zhou, M., Sigmund, O., & Zhang, X. S. (2021). A comprehensive review of educational articles on structural and multidisciplinary optimization. *Structural and Multidisciplinary Optimization*, *64*(5), 2827–2880. https://doi.org/10.1007/s00158-021-03050-7

Xu, S., Cai, Y., & Cheng, G. (2010). Volume preserving nonlinear density filter based on heaviside functions. *Structural and Multidisciplinary Optimization*, *41*(4), 495–505. https://doi.org/10.1007/s00158-009-0452-7