Technical University of Denmark

Interactive topology optimization

Aage, Niels; Nobel-Jørgensen, Morten; Andreasen, Casper Schousboe; Sigmund, Ole

Publication date:
2012

Document Version
Publisher's PDF, also known as Version of record

Interactive topology optimization

Niels Aage*, **Morten Nobel-Jørgensen, Casper S. Andreasen, Ole Sigmund**

Department of Mechanical Engineering, Solid Mechanics,
Technical University of Denmark, Nils Koppels Alle, B.404,
DK-2800 Kgs. Lyngby, Denmark
*E-mail: naa@mek.dtu.dk

Keywords: interactive topology optimization, smartphones, multigrid preconditioned CG.

ABSTRACT

The motivation for this work originates from the ever increasing usage of smartphones and other handheld devices such as tablet computers. Combined with the fast growth in computational power, these devices now constitutes platforms on which advanced engineering analysis can be performed. In this manuscript we present an App for mobile and web-based platforms, which is capable of performing interactive 2D topology optimization for the minimum compliance problem. We expect the App to be relevant for engineering, architectural and design education.

Topology optimization is an iterative design method used to optimize material layouts of a variety of physical systems [3]. Usually the method is applied using a passive problem description, such that the optimization is performed for a fixed set of loads, supports and optimization parameters. An example of a passive topology optimization applet can be found on the TopOpt-groups homepage (www.topopt.dtu.dk). This applet has been hosted since 1999 and has been used extensively with over 200.000 runs by more than 13.000 unique users. Opposed to the passive approach to topology optimization, the main idea of the interactive topology optimization App is to allow the user to modify supports, loads and optimization parameters on the fly and see the impact immediately.

The developed App solves the minimum compliance problem from linear elasticity, which can be stated as a mathematical programme as seen below

$$
\begin{aligned}
\min_{\rho \in R^n} \quad & \phi = u^T F \\
\text{s.t.} \quad & K(\rho) u = F \\
& \textstyle\sum_{i=1}^{n} \rho_i V_i - V^* \leq 0 \\
& 0 < x_i \leq 1, \quad i = 1, ..., n
\end{aligned} \tag{1}
$$

where ρ are the densities, u the displacements, F the load vector and $K(\rho) = \sum_e \rho_e^p K_e^0$, where $p = 3$ is a penalization factor and K_e^0 is a reference element stiffness matrix. This design interpolation is known as the density approach or the SIMP method [3]. The three constraints represent the finite element discretization of the Hooke's law [5], a restriction on the available material and box constraints for the design variables, respectively. In this work the optimization problem is solved as a nested problem using a simple optimality criterion to update the densities [3].

The main goal in the development of an interactive topology optimization App, is to write a fast and efficient optimization code such that the user will experience that the structure will respond quickly, i.e. become alive, when loads or supports are moved around, added or removed. As a secondary goal we wish to make the App available for multiple platforms, e.g. iOS, Android and Web. The cross-platform portability is achieved by using the game engine Unity3D (www.unity3d.com) for constructing the

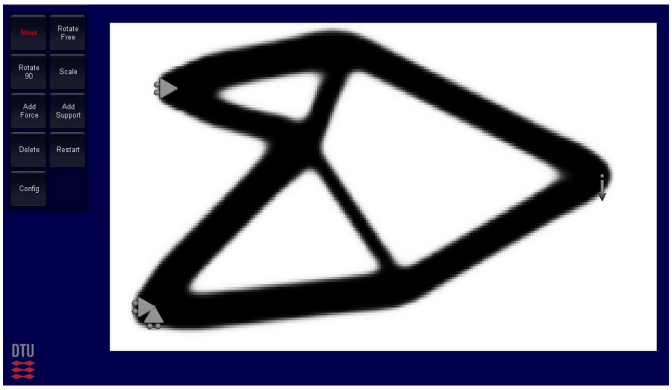

Figure 1: Screenshot of the developed App.

GUI and to deal with real-time user-interaction. This in turn requires that all code is written in C#. The cross platform requirement means that we cannot fully utilize optimized dense matrix libraries such as BLAS/LAPACK. In order to obtain a efficient solution to each design cycle we use a multigrid preconditioned conjugate gradient (MG-PCG) method for solving the state problem. The multigrid method is applied in a V-cycle using Jacobi iterations for smoothing and a sparse LDL factorization to solve the state equation on the coarsest level. We have tested both the Galerkin approach as well as an implementation with full-weighting for prolongations and injection for restrictions. Both methods shows good convergence and by utilizing that the minimum compliance problem and the conjugate gradients method, minimizes the same functional, we can drastically reduce the number of iterations by applying the PCG convergence criteria as given in [2]. To further reduce computational costs we use ideas from re-analysis [1], to reduce the number of times the coarse level matrix is factorized. Finally, we use the multiresolution design methodology from [4], which allows us to have e.g. four design variables per finite element and thus a finer representation of the design. For regularization we employ both sensitivity and density filtering to avoid checkerboards.

The current state of the App can produce a framerate of four design cycles per second on a finite element mesh of 40×30 on an iPad2. A screenshot can be seen in figure 1. Future work on the optimization kernel concerns hardware specific models, e.g. utilizing Apple's iOS implementation of BLAS/LAPACK, and further tuning of the multigrid solver.

References

[1] O. Amir and O. Sigmund. On reducing computational effort in topology optimization: how far can we go? *Structural and Multidisciplinary Optimization*, 44:25–29, 2011.

[2] M. Arioli. A stopping criterion for the conjugate gradient algorithm in a finite element method framework. *Numerische Mathematik*, 97:1–24, 2004.

[3] M.P. Bendsøe and O. Sigmund. *Topology Optimization; Theory, Methods and Applications*. Springer Verlag Berlin Heidelberg New York, 2nd edition, 2004.

[4] T. H. Nguyen, G. H. Paulino, J. Song, and C. H. Le. A computational paradigm for multiresolution topology optimization (mtop). *Struct Multidisc Optim*, 41:525–539, 2010.

[5] O. C. Zienkiewicz and R. L. Taylor. *Finite Element Method: (parts 1-3)*. Butterworth-Heinemann, fifth edition, 2000.