Large-scale structural design optimization for a thin plate

Ning Li and Jiayao Yan
University of California San Diego, La Jolla, CA, 92093

OpenMDAO, an open-source Python framework, has become mature and efficient for the design optimization of engineering systems, even in the presence of large numbers of design variables and constraints. However, OpenMDAO doesn't have an efficient finite element solver for structural design optimization. On the other hand, tIGAr is a powerful Python library for isogeometric analysis (IGA). Integration of OpenMDAO and tIGAr can be beneficial to engineers who work on structural optimization. We present a large-scale structural design optimization for a thin plate using OpenMDAO and tIGAr to test their compatibility.We first minimize the compliance of the plate with respect to the thickness of a cantilever thin plate subjected to different volumes constraints, and then change the boundary conditions to see how volume of the plate and the boundary conditions affect the optimization results.

I. Introduction

The approach of Large-scale design optimization is well-suited for handling problems with a large variable size and great complexity. Large-scale design optimization problems involve hundreds or more design variables. A discipline that is frequently considered in large-scale design optimization applications is structures. Although the finite element method (FEM) is the dominant structural analysis solution procedure, it is originally defined on the meshes of data points which implies its accuracy depends significantly on the quality of the mesh. However, great geometric approximation error can be introduced using a lower order mesh element, while if higher order elements are employed, more details of the geometry may be preserved, the computational time can also increase consequently. Isogeometric analysis (IGA) can be hired to solve such problems the conventional FEM has. IGA creates geometry based on B-splines and NURBS which do not require connection between nodes of the simulation domain, like mesh, instead relies on the interaction of each node with all its neighbouring nodes. Thus, in IGA, the domain can be represented exactly so that no geometry approximation error is introduced.

This project works on the integration of the large-scale design optimization and isogeometric analysis, using OpenMDAO and tIGAr frameworks. OpenMDAO is an open-source Python framework that simplifies multidisciplinary optimization of large-scale problems—those that have at least hundreds of design variables [1]. To overcome the curse of dimensionality in optimization, OpenMDAO uses gradient-based methods with analytically-computed derivatives. However, OpenMDAO doesn't have a component that can be used to do finite element analysis and all the gradients must be calculated manually, which is time-consuming and demands a great amount of human effort. On the other hand, tIGAr is a library that performs isogeometric analysis (IGA) by using an open-source finite element automation software FEniCS[2]. The FEniCS Project covers some open-source software elements, including Unified Form Language (UFL)[3]. UFL can be used to calculate the Gateaux derivatives automatically to get rid of OpenMDAO's tedious manual effort in gradient calculation[4].

Since we are currently working on an automatic shell element mesh generation method and will further apply our generated mesh to structural design optimization of an aircraft structure. Aircraft structures are commonly modeled using shell elements, and they require high-quality meshes to model accurately. Thus, the integration of OpenMDAO and tGIAr provides great computational ability for our research. As a beginning, it can be beneficial to test the compatibility of OpenMDAO and tIGAr with a simple exmaple. Thus in this project, we perform an structual design optimization of a thin plate using both OpenMDAO and tGIAr.

II. Problem Formulation

The problem can be split into structural analysis and optimization setup. In tIGAr, we complete the biharmonic model of a thin plate and calculate all the related gradients, while in OpenMDAO the optimization problem is formulated.

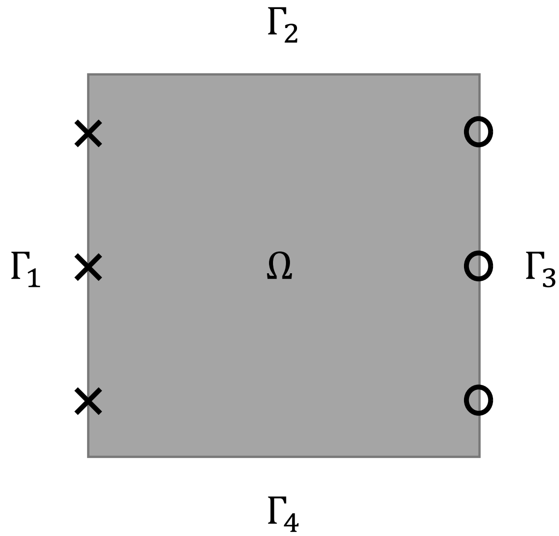


Fig. 1 The boundary is fixed along the left edge Γ_1, denoted by the cross mark, while an uniformly distributed force is applied on the right edge Γ_3. No external forces or bending moment are applied to the upper and lower edges, Γ_2 and Γ_4.

A. Structural Analysis

For simplicity, in the structural analysis of the thin plate, we assume each cross section of the plate remains plane and is orthogonal to the internal neutral plane after deformation. Thus, the shear strain energy is much smaller than the bending strain energy and we can use the isotropic quasistatic Kirchhoff-Love plate model to describe the behavior of the plate,

$$\nabla^2\nabla^2 w = 0, \tag{1}$$

which is known as the biharmonic equation where w is the deflection of the plate. The plate is chosen to be a square with its left edge Γ_1 fixed and an unifromly distributed force applied on the right edge Γ_3. The other two edges Γ_2 and Γ_4 are 'free' boundaries with no applied external shear force or bending moment as shown in Fig. 1. Thus the strong form of the problem is constructed as:

$$\begin{aligned} & \nabla^2\nabla^2 w = 0, \\ \textbf{Boundary Condition} \quad & w(x) = 0,\ w_{,x}(x) = 0 \text{ on } \Gamma_1 \\ & EIw_{,xx}(x) = 0,\ EIw_{,xxx}(x) = h \text{ on } \Gamma_3 \\ & EIw_{,xx}(x) = 0,\ EIw_{,xxx}(x) = 0 \text{ on } \Gamma_2 \text{ and } \Gamma_4, \end{aligned} \tag{2}$$

where E is the Young's modulus and I is the second moment of area, which in the plate case can be calculate by$I = \frac{bh^3}{12}$, where b is the length of the plate and h is the thickness. As we set the $\frac{b}{12}$ as the unit length, I can be simply computed as h^3. The weak form of the biharmonic equation can be expressed as:

$$\int_\Omega \Delta w \Delta v d\,\mathbf{\Omega} + \int_{\partial\Omega} \boldsymbol{n} \cdot \nabla(\nabla \cdot (\nabla w))v\ d\mathbf{\Gamma} - \int_{\partial\Omega} \nabla \cdot (\nabla w)(\nabla v \cdot \boldsymbol{n})\ d\mathbf{\Gamma} = 0 \tag{3}$$

where Ω is the entire domain, $\partial\Omega$ is the boundary of the domain, v is the trial function and $\boldsymbol{n}$ is the unit vector normal to the boundary of the domain. Weakly enforcing the boundary conditions, we get

$$\int_\Omega \Delta w \Delta v d\,\mathbf{\Omega} + \int_{\Gamma_3} \boldsymbol{n} \cdot \nabla(\nabla \cdot (\nabla w))v\ d\mathbf{\Gamma} = 0. \tag{4}$$

Approximating the domain with B-splines and thickness in discontinuous Garlerkin space and plugging in the weak form in IGA, the finite element analysis is completed.

B. Optimization Setup

The optimization problem is defined in Eqn. 5. We define the objective function as minimizing the compliance of the plate, which is denoted as the displacement on the right edge with respect the the thickness of each element in the mesh. The total volume of the plate is constrained to be a constant and the upper and lower bounds for the thickness are declared.

$$\begin{aligned} \textbf{minimize} \quad & \int_{\Gamma_3} \frac{1}{2} w^2 d\boldsymbol{\Gamma} \\ \textbf{with respect to} \quad & 0.01 < t < 10 \\ \textbf{subject to} \quad & \int_{\Omega} t d\boldsymbol{\Omega} = C \end{aligned} \tag{5}$$

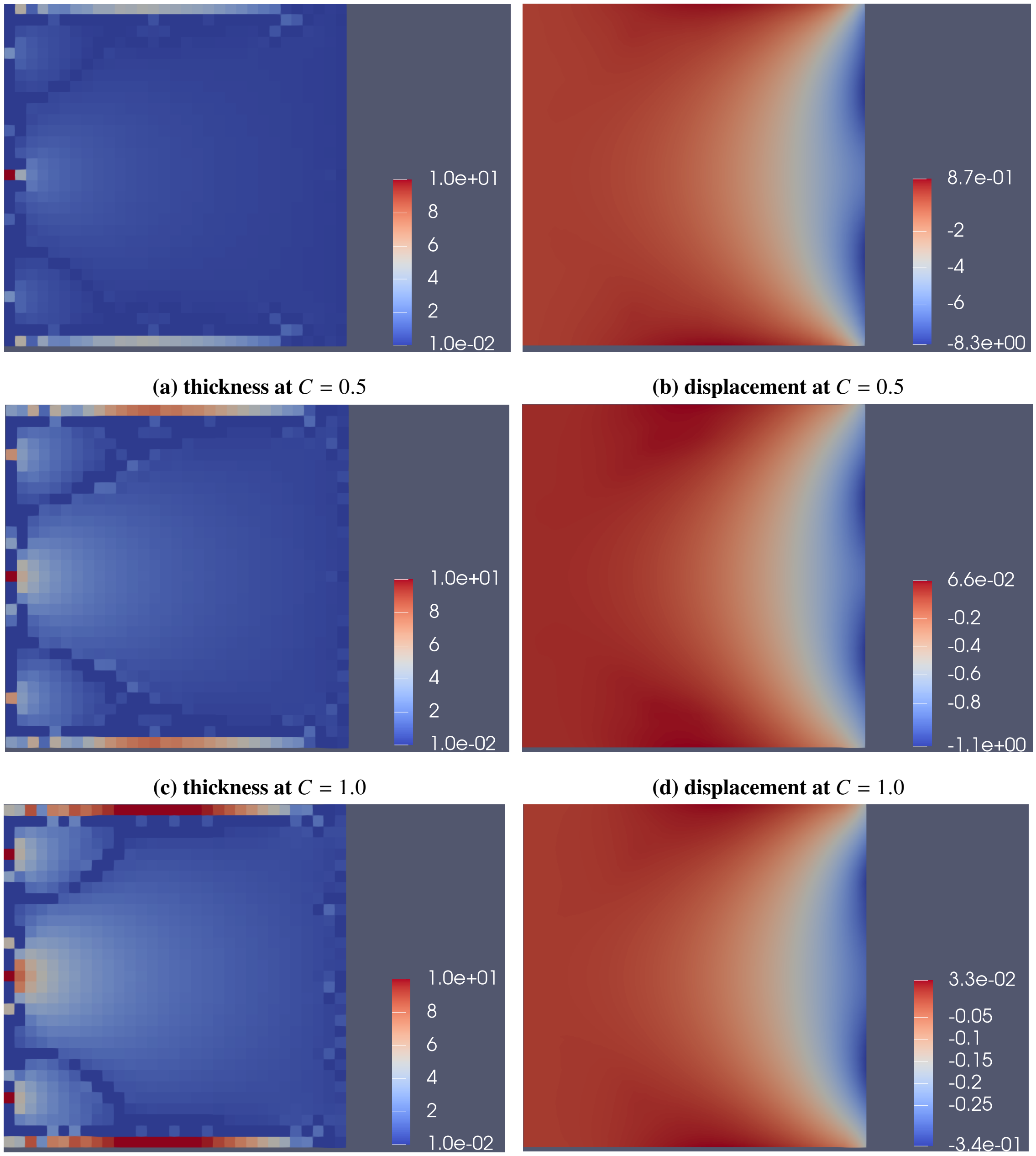
where C is the volume which is changed to different values to learn the effect the volume has on the optimization results.

III. Results and Discussion

In this section, we discuss the results with different volume constants and different boundary conditions. The total number of elements used in each direction is 31. The number is chosen to be odd because the solution is axisymmetric about the horizontal middle plane and if we choose an even number, the result is asymmetric. Also if the number is chosen to be larger than 40, it takes around 30 minutes to finish the optimization. To get a relatively accurate solution in a comparatively short time, 31 is chosen. First we use $C = 0.5$, $C = 1.0$ and $C = 1.5$ to see how different volume affects the distribution of the materials. Then, we fix the upper and lower edge of the square plate to see what happens when the boundary condition is changed.

A. Design Optimization with Different Volume

Initially, we only perform the optimization with volume being 0.5, however as shown in Fig. 2a, the thickness distribution of plate doesn't behave as we expect. Intuitively, the thickness should be the largest at the fixed end and monotonically decrease along horizontal direction to the end where the uniformly distributed force is applied. Although the optimization results show that generally the thickness decreases horizontally from the left to right edge, especially in the middle region, there are some elements at the upper-left and bottom-left corners that has significantly smaller thickness than their neighbouring elements. This is surprising as when we do the beam optimization with left end fixed and load applied on the other end, the optimal thickness distribution gives us a perfect decreasing curve. However, the displacement profile at $C = 0.5$ in Fig. 2b to some extent makes sense, since instead of having monotonically positive or negative displacements, it has mixed displacement profile in that near the right edge, the displacement is negative while on the rest elements the displacement is positive. Intuitively by employing the mixed displacement profile, the displacement at the left end won't accumulate to the right end. We guess the volume may be the reason for this results, so we do another two optimizations with increased volume of 1.0 and 1.5 as shown in Figs. 2c, 2d, 2e, and 2f. The results still show the similar patterns to the case when $C = 0.5$, especially the displacement profile. Although the magnitude of displacement may be different for $C = 0.5$, $C = 1.0$, and $C = 1.5$, the overall distributions of the thickness are pretty similar. One possible reason for the different results of beam and plate optimization is that the boundary conditions of each row of elements in the plate cannot be viewed as the same, since the central rows of elements are supported by the neighbouring rows of elements, while the upper and lower edge are only supported from one side. We suppose that if the plate is much longer vertically than horizontally, we are able to see monotonically decreasing thickness distributions in the central regions, just like what we have in the beam optimization.



(a) thickness at C = 0.5 **(b) displacement at** C = 0.5

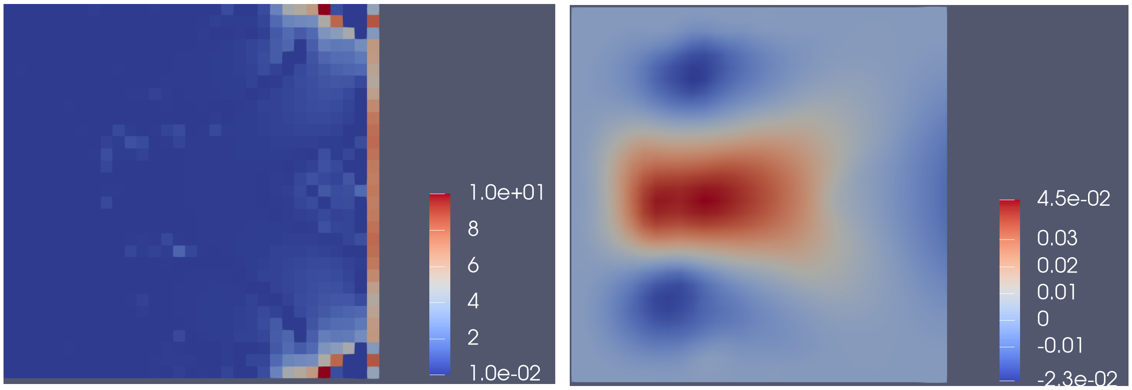
(c) thickness at C = 1.0 **(d) displacement at** C = 1.0

(e) thickness at C = 1.5 **(f) displacement at** C = 1.5

Fig. 2 Thickness and displacement profiles for C = 0.5, 1.0, 1.5.

B. Design Optimization with Different Boundary Condition

In this section, we additionally fix the upper and lower boundary Γ_2 and Γ_4 to see if we are able to get a monotonic decreasing thickness from the fixed boundary to the boundary where the force is applied. As shown in Figs. 3a and 3b, unfortunately we still cannot achieve expected profile and the thickness profile is not even exactly axis-symmetric. One possible reason for this is that due to more fixed boundaries, the symmetry of the solution gets more complex and thus we may need to refine the mesh to get a better result. For some elements in the center of the plate which has a larger thickness, we believe it is because the optimization tends to generate a result mixed of positive and negative displacements to consequently decrease the displacement at the boundary where the force is applied.



(a) thickness at $C = 0.5$ with three fixed boundary **(b) displacement at $C = 0.5$ with three fixed boundary**

Fig. 3 Thickness and displacement profiles with additional two boundaries fixed.

IV. Conclusion and Future work

The integration of OpenMDAO and tIGAr is tested and we can now perform structural design optimization of a plate structure. Although some features of the optimization results remain unclear, this project still proves the compatibility of OpenMDAO and tIGAr. For the future study, we can improve results of the project if we can:

1) Change the optimizer to SNOPT to allow more elements;

2) Import mesh from external sources like the generated meshes or B-splines surfaces from my research project;

3) Set thickness in a continuous Garlerkin space or use B-splines to get a smoother profile.

References

[1] Gray, J. S., Hwang, J. T., Martins, J. R., Moore, K. T., and Naylor, B. A., "OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization," *Structural and Multidisciplinary Optimization*, 2019, pp. 1–30.

[2] Kamensky, D., and Bazilevs, Y., "tIGAr: Automating isogeometric analysis with FEniCS," *Computer Methods in Applied Mechanics and Engineering*, Vol. 344, 2019, pp. 477–498. https://doi.org/https://doi.org/10.1016/j.cma.2018.10.002.

[3] Logg, A., Mardal, K.-A., and Wells, G., *Automated solution of differential equations by the finite element method: The FEniCS book*, Vol. 84, Springer Science & Business Media, 2012.

[4] Alnæs, M. S., Logg, A., Ølgaard, K. B., Rognes, M. E., and Wells, G. N., "Unified form language: A domain-specific language for weak formulations of partial differential equations," *ACM Transactions on Mathematical Software (TOMS)*, Vol. 40, No. 2, 2014, pp. 1–37.