Engineered Model Simplification for Simulation Based Structural Design

Mingdong Zhou and Michael Yu Wang

The Chinese University of Hong Kong
mdzhou@mae.cuhk.edu.hk yuwang@mae.cuhk.edu.hk

ABSTRACT

Simulation based structural design usually involves frequent conversions between implicit and explicit models. In this paper, a fast and robust mesh simplification method is proposed to improve the efficiency of implicit evaluation of large size engineered models. The mesh is simplified via edge collapse with a metric consisting of edge length and curvature information. A progressive multi-pass strategy makes the simplification reasonable and effective. Topology consistency is maintained by a validity check before each primitive edge removed. Most importantly, this method guarantees the practical utility of simplified model in downstream simulation based structural design. Satisfactory results of mesh simplification and its application are provided to demonstrate the capability.

Keywords: engineered model, mesh simplification, simulation based structural design.
DOI: 10.3722/cadaps.2012.87-94

1 INTRODUCTION

In modern product development, when market has put more emphasis on high performance, low cost and short lead time, it is realized that integration of design, analysis and optimization in a consistent and automated way would definitely promote product's competitiveness. There is an important class of design process, namely simulation based structural design, which refers to the optimization of structural product according to its performance requirements. This process aims to minimize the use of physical prototype, and has been widely applied in areas of aerospace, automotive and steelmaking.

Structural analysis is the core part in simulation based structural design and usually done by conducting finite element analysis on computer aided designed model. Numerical methods, especially the non-remeshing finite element method such as ersatz material approach [1,3,22] and extend finite element method (XFEM) [4], provide efficiency and simplicity. However, these methods require implicit model as input, while most engineered models are represented explicitly by using boundary representation. Hence, frequent conversions between implicit and explicit models are inevitable. Moreover, for a simulation based design process, it is quite common to have very large number of triangles to represent a complex engineered model after high resolution analysis. During the conversion from explicit to implicit model, since primitive operation of calculating the nearest distance from an arbitrary point to a polygon is expensive, the size of model affects the performance of the design process remarkably. Therefore, to reduce model size without losing its engineering signification is crucial for simulation based design in promoting computational efficiency.

At present, triangle meshes are still the most versatile representation for free-form geometric entities. Mainstream approaches of simplification of triangle mesh are classified into volume based and surface based methods.

Volume based methods leverage spatial model as intermediate agency to simplify the surface model. One of the earliest volume based simplification algorithm is proposed by He et al. [8]. In their work, the object is firstly sampled and filtered into multi-resolution volume buffers, and triangle mesh hierarchy is then generated using marching cube method. An octree based mesh decimation method can be found in [18]. Recently, Binary Space Partition (BSP) tree has been explored for volume based simplification. In [19], it presents a shape approximation method using BSP with bounded approximation error. In [12], Huang et al. successively propose a volume and complexity bounded solid simplification technique for models represented by BSP. Although volume based method can produce satisfactory simplification results in terms of geometric complexity, it generally takes higher computational complexity than surface based approach.

Among various surface based algorithms, vertex clustering, resampling and incremental decimation are representative techniques. Vertex clustering refers to the process of clustering vertices into groups and replacing each group by one representative vertex according to different metrics [10,13,17]. It is efficient, but maintaining topology is difficult so that it always produces non-manifold mesh. Resampling approach firstly generates new samples that are freely distributed over the original surface, and then new mesh is constructed by connecting the samples in a special structure [6,7,21]. However, alias error may become disastrous if the sampling pattern is not aligned to features. Incremental decimation methods, specifically the edge collapse method, can take arbitrary user-defined criteria into account to define the best edge candidate to be removed, as well as to determine the location of contracted vertices [9, 10]. Topology can be preserved by carefully choosing candidate edge to collapse without incurring any topological errors [11,23]. However, traditional approaches suffer two problems. One is the inefficient issue for simplifying large size model using one pass strategy, in which a priority queue is firstly set up based on certain metric measure and edge candidates in the queue are then simplified in sequential order. The other issue lies in that, if the collapse ratio (the percentage of decimation) is not carefully set, it may excessively simplify a certain region, leaving other parts, which expected to be simplified, untouched.

In this paper, we present a novel progressive multi-pass mesh simplification method based on edge collapse. This method not only fixes the abovementioned problems but also is feature sensitive and topological errorless. Besides, our objective is to shorten the design process without at the expense of losing accuracy. Hence the highly versatility and effectiveness of edge collapse method make itself a good auxiliary candidate for practical simulation based design. We underscore the importance of simplification strategy, because it impacts notably on the efficiency of execution. Moreover, because the simplified model will be served in downstream post-processing like evaluation of implicit field, it is a hard constraint that no topological errors, such as triangle overlapping or flipping, and degenerated faces, should occur. By carefully imposing topological validity check before each collapse operation, we are able to guarantee the output mesh is closed 2-manifold of errorless topology. Consequently, simulation based structural design can benefit by the proposed algorithm of fast response and pleasing result.

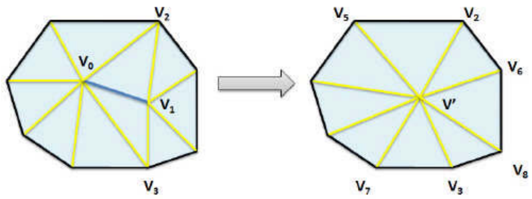

Fig. 1: Primitive operation of edge collapse: the blue edge spanned by v_0 and v_1 is collapsed to new vertex v' ; the yellow edges are affected ones whose weight must be re-evaluated after collapse.

2 PROGRASSIVE MULTI-PASS SIMPLIFICATION

2.1 Algorithm

For the majority of engineered models, the shape is largely dominated by flat or smooth patches, while the special characteristics and geometric meaning are captured by a few typical features, such as edges and corners. Notably, inherent property, like curvature, at feature region is much higher than other regions. In order to obtain a compact model without losing engineering significance, fine details must be retained at feature region, but the other parts should be simplified as much as possible. Therefore, the following metric combining edge length and curvature information is adopted for simplification:

$$W_i = \left(\frac{\kappa_{i_0} + \kappa_{i_1}}{\kappa_{\max}} + \alpha\right) \cdot l_i$$

where $\kappa_{i_0}, \kappa_{i_1}$ denoting the mean curvature at vertices share the edge with index i, $\kappa_{\max}$ the maximum mean curvature over the mesh, α a dimensional parameter, and l_i the length of edge. Mean curvature at vertices can be approximated using the method proposed in [15].

The rationale behind this measure lies in following considerations. On the one hand, for vertices at flat regions, where the first part of the metric vanishes, the non-zero value parameter α will annotate each edge with a weight according to its length. On the other hand, carefully setting the dimensional parameter α will balance the weight of edge length and curvature information at non-flat regions. In fact, setting the value of dimensional parameter α depends on application, and no universal optimal value exists. The effect of different parameter setting is studied in next section. Note that, by setting α equals to zero, the same metric as in [23] can be obtained.

Traditional edge-collapse based simplification methods usually start with building a priority queue based on certain metric, and then sequentially collapse items in the queue from head to tail. However, such one-pass simplification suffers two pitfalls. Firstly, collapsing one edge requires to re-evaluate the weight of nearby affected edges (Fig.1), and to reorder their positions in the queue. The complexity of repeatedly inserting the K edges into the queue is prohibitive, because each collapse will costs up to $O(K \log N)$ searching. N is number of edge in currently priority queue and usually very large at the beginning of simplification process. Secondly, if collapse ratio is not properly set, it always excessively simplify particular regions, meanwhile other parts which are expected to be simplified remain unchanged. Fig. 1 shows the primitive operation of edge collapse used in this paper. The candidate edge v_0 and v_1 is collapse into a new vertex at the middle of the edge.

In order to avoid the abovementioned problems, a progressive multi-pass strategy is adopted. It periodically collapses edges in priority queue, until the total number of triangles reaches the collapse ratio or no suitable candidate edge (topological check validated) is available in the queue. Whenever collapse one edge, the affected edges around it are frozen temporarily, and will not be considered for decimation until next pass rather than immediately re-evaluation and re-insertion into the queue. Hence it avoids repeatedly searching the queue. Performing one quick sort at beginning of each pass only takes $O(N \log N)$ complexity, and N is decreased dramatically for each pass. In practice, four passes is enough for most of our testing models.

The collapse ratio of each pass must be also carefully set to amortize the chance of simplification over all regions of the model. We found that setting the collapse ratio to a relative large initial value, and then reducing it gradually worked well. By doing so, different regions can be simplified simultaneously at first, and then refined iteratively in later passes.

Triangle flipping or overlapping is a common problem in edge-collapse based method if one does not handle topology properly. It is crucial to be avoided for our problem because evaluation of implicit field from explicit model heavily relies on the quality of surface mesh. Specifically, the mesh must be closed, 2-manifold, topological errorless and non-degenerated. To guarantee the simplified mesh satisfying these requirements, we perform following edge validity check before any collapse operation, the idea directly comes from [11, 23].

1) Non-manifold check: $v_5 \neq v_6$, $v_7 \neq v_8$, and if any vertex connects to both v_0 and v_1, there must be a triangle expanded by these three vertices.
2) Normal flipping check: the angle of normal deviation of all the triangles adjacent to v_0 and v_1 after collapse must be smaller than a threshold. Note that the smaller the threshold, the less chance of collapse at feature region.
3) Triangle quality control: triangle quality is measured by:

$$Q = \frac{4\sqrt{3}A}{L_0^2 + L_1^2 + L_2^2},$$

where A the triangle area after collapse and L_i the edge length. When $Q \to 1$, the triangle approaches to an equilateral triangle, and when $Q \to 0$, degenerated one appears. Hence the quality measure Q of each affected triangle after collapse must be larger than some threshold $0 < \mu < 1$. For our problem, since we only need to guarantee that no degenerate face exists, this criterion seems loose as long as the normal of triangles are computable and consistent.

2.2 EXPERIMENTAL RESULTS

Tab. 1 lists the pseudocode of our multi-pass simplification algorithm. We applied it to several engineered models and one standard digital shape on PC with 4GB RAM and Intel(R) Core(TM)2 Quad CPU at 2.66GHz.

Fig. 2 shows the multi-passes simplification process, during which the rocket arm model is simplified iteratively. In order to give a clear sense of choosing dimensional parameters, Fig. 3 illustrates the effect of simplification with same collapse ratio, but with different setting of α. For small values, simplification tends to retain features and focus on simplifying the flat regions. But as the value increased, it behaves like a uniform remeshing process. In our experiments, we have found that setting α between 0.0001 to 0.001 could produce pleasing results for most of engineering models, which initially contains flat patches of densely mesh. Nevertheless, properly setting of α depends on application, and no universal optimal setting exists.

We also compared our algorithm with the quadratic edge collapse decimation method in MeshLab, an open source system of editing 3D triangle mesh [14]. Tab. 2 records detailed statistics for each

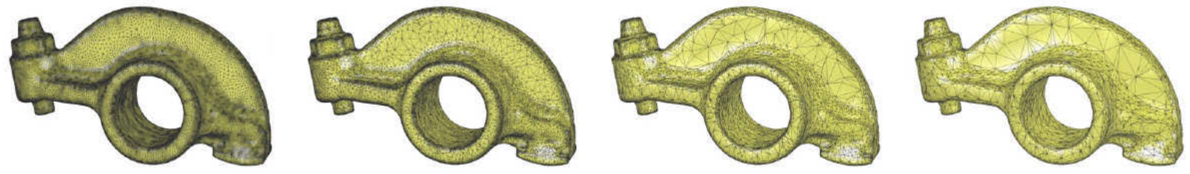

Fig. 2: Progressive simplification of a rocket arm model (left) in three passes.

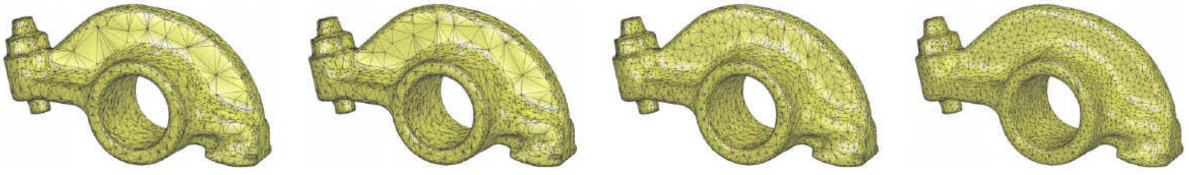

Fig. 3: Dimensional parameter α (from left to right) : 0, 0.0001, 0.1,and 1.

Algorithm multi-pass simplification

```
Input: closed 2-manifold triangle mesh, array of collapse ratio, iterative number total_num
iter_num=0
while iter_num is smaller than total_num
    compute mean curvature for each vertex, and length for each edge;
    build a priority queue Q based on proposed measure with smaller values at head;
    while Q is not empty and the collapse ratio at current pass is not satisfied
        Choose the head item E as edge candidate;
        if E is frozen
            then remove E from Q; continue;
        else if E violate validity check
            then remove E from Q; continue;
        else
            collapse E, freeze all effected edge, and remove E from Q ;
        end if
    end while
    iter_num++
end while
```

Tab. 1: Pseudocode of multi-pass simplification algorithm.

Model	*Algorithm*	*Initial Face No.*	*Final Face No.*	*Execution Time (s)*
Rocket arm	*MeshLab*	*180,792*	*17,540*	*4.10*
	Our Method	*180,792*	*17,540*	*1.79*
Pulley	*MeshLab*	*587,344*	*39,490*	*13.55*
	Our Method	*587,344*	*39,490*	*5.18*
Oil pump	*MeshLab*	*1,140,048*	*82,118*	*26.40*
	Our Method	*1,140,048*	*82,118*	*13.46*
Dragon	*MeshLab*	*49,416*	*17,526*	*0.95*
	Our Method	*49,416*	*17,526*	*1.45*

Tab. 2: Statistics of experimental results.

model, and Fig. 4-6 show the experimental results. The dimensional parameter is set to be 0.0001 for all test cases. In fact, our method is generally 2-3 times faster than that in MeshLab for large size model. It also essentially guarantees no topological errors, such as triangle overlapping, flipping or degenerate faces. Reasonable simplification can be carried out for engineered models, which leaves coarse mesh at flat region and dense resolution at feature regions. All these merits are promising for simulation based structural design.

3 APPLICATION IN SIMULATION BASED STRUCTURAL DESIGN

To further illustrate the practicability of our novel simplification algorithm, we apply it in a structural optimization framework using an interface tracing algorithm of level set method, which is an extension to that in Bargteil's work [2].

The optimization process starts with an initial structural configuration, a triangle mesh model, according to which the design domain is adaptively subdivided into an octree grid. Each node in the tree is annotated with signed-distance value that computed directly from the surface mesh. After the implicit model obtained, a finite element analysis procedure is evoked to evaluate the performance with respect to specific objective and constraints. If it is not optimal, the structural interface evolves via a semi-Lagrange scheme [20], which is followed by surface extraction from the updated implicit field using a marching tetrahedron method [5]. However, the major drawback of marching cube type method is that it produces huge number of faces. Since the distance evaluation bounds up directly and tightly to the model size, the extracted model will drags down overall efficiency of design process prohibitively. Therefore, the significance of applying mesh simplification afterwards is straightforward. Consequently, the implicit field can be updated efficiently based on the simplified model, which refers to a precise re-initialization process [16], so that optimization can continue.

Nevertheless, efficiency must be improved without at the expense of losing computational accuracy. Fortunately, our simplification method is actually a feature preserving and topological errorless technique, which can simplify large size model reasonably by preserving engineering signification for each step of optimization. Therefore, it is suitable to apply this method to any simulation based algorithm which contains heavy implicit and explicit model conversions.

Fig. 7 shows the initial, intermediate and optimal structural configurations of compliance minimization problem of linear elastic continuum structure with volume constraint. With help of our mesh simplification, each optimization iteration costs about 15 seconds, which in average doubles the speed of that without simplification. The resolution of the octree used in our example is of 64 x 64 x 64. All the related parameters in this method are tuned by experiment.

4 CONCLUSION

This progressive multi-pass simplification method is efficient and effective, especially applicable to large size engineering mesh models. Engineering significance is well preserved and no topological errors occur. These merits guarantee the practical utility for simplified models in downstream post-processing. Satisfactory results demonstrate its capability of serving as an auxiliary module in simulation based design to accelerate design process. However, we have not concerned about the quality of surface and boundary deviation in simplification, which plays a vital role in other optimization topic, such as stress and bucking related problems. Our future work may further study the overall impact on design of using this simplification method, as well as to take volume preserving metric into account.

REFERENCES

[1] Allaire, G.; Gournay, F.; Jouve, F.; Toader, A.-M.: Structural optimization using topological and shape sensitivity via a level set method, Control and Cybernetics, 34, 2005, 59-80.

Fig. 4: Pulley, (left) original solid and mesh model; (middle) MeshLab; (right) our method.

[2] Bargteil, A.-W.; Goktekin, T.-G.; O'Brien, J.-F.; Strain, J.-A.: A semi-Lagrangian contouring method for fluid simulation, ACM Transactions on Graphics, 25(1), 2006. DOI:10.1145/1122501.1122503

[3] Bendsoe, M.-P.: Optimal shape design as a material distribution problem, Structural Optimization 1, 1989, 193-202. DOI:10.1007/BF01650949

[4] Belytschko, T., Black, T.: Elastic crack growth in finite elements with minimal remeshing, International Journal for Numerical Methods in Engineering, 45(5), 1999, 601-20. DOI:10.1002/(SICI)1097-0207(19990620)45:5<601::AID-NME598>3.0.CO;2-S

[5] Bloomenthal, J.: An implicit surface polygonizer, In Graphics Gems IV, Academic Press Professional, Inc., San Diego,CA, 1994, 324–349.

[6] Botsch, M.; Pauly, M.; Kobbelt, L.; Alliez, P.; Lévy, B.: Geometric modeling based on polygonal meshes, Eurographics Tutorial, 2008

[7] Eck, M.; DeRose, T.; Duchamp, T.; Hoppe, H.; Lounsbery, M.; Stuetzle, W.: Multiresolution analysis of arbitrary meshes, In Proc. of ACM SIGGRAPH 95, pp.173–182, 1995.

[8] He, T.; Hong, L.; Kaufman, A.; Varshney, A.; Wang, S.: Voxel based object simplification, In VIS '95: Proceedings of the 6th conference on Visualization '95, page 296, Washington, DC, USA, 1995.

[9] Garland, M.; Heckbert, P.: Simplifying surfaces with color and texture using quadric error metrics, In IEEE Visualization Conference Proceedings, 1998.

[10] Garland, M.; Heckbert, P.: Surface simplification using quadric error metrics, In Proc. of ACM SIGGRAPH 97, pages 209–216, 1997.

[11] Hoppe, H.; DeRose, T.; Duchamp, T.; McDonald, J.; Stuetzle, W.: Mesh optimization, ACM SIGGRAPH 1993 Conference Proceedings, 19-26.

[12] Huang, P.; Wang, C.- C.: Volume and complexity bounded simplification of solid model represented by binary space partition, ACM Symposium on Solid and Physical Modeling 2010, pp.177-182, Haifa, Israel, September 1-3, 2010.

[13] Lindstrom, P.: Out-of-core simplification of large polygonal models, In Proc. of ACM SIGGRAPH 00, pp. 259–262, 2000.

[14] MeshLab, http://meshlab.sourceforge.net/

[15] Meyer, M.; Desbrun, M.; Schroder, P.; Barr, A.: Discrete differential-geometry operators for triangulated 2-manifolds, Visualization and Mathematics III, H.-C. Hege and K. Polthier, eds., Springer-Verlag, 2003.

[16] Osher, S.J.; Fedkiw, R.P.: Level Set Methods and Dynamic Implicit Surfaces, Springer, 1 edition October 31, 2002

[17] Rossignac, J.; Borrel, P.: Multi-resolution 3D approximations for rendering complex scenes, In Geometric Modeling in Computer Graphics , pp. 455–465. Springer Verlag, 1993.

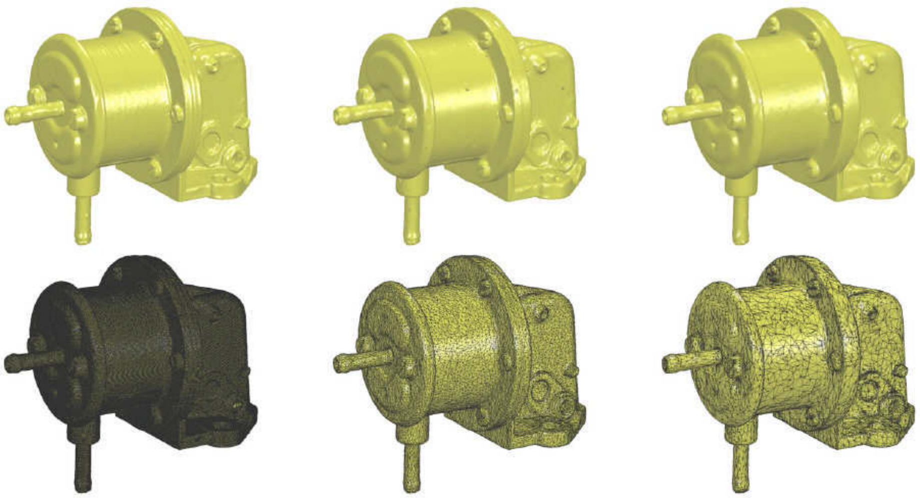

Fig. 5: Oil pump, (left) original solid and mesh model; (middle) MeshLab, black spots in upper figure are due to topological errors; (right) our method.

[18] Shekhar, R.; Fayyad, E.; Yagel R.; Cornhill, J.- F.: Octree-based decimation of marching cubes, Surfaces, In Proceedings of the 7th conference on Visualization '96, 335–342, Los Alamitos, CA, USA, 1996.
[19] Stoll, C.; Seidel, H.-P.; and Alexa, M.; Bsp shapes; In SMI '06: Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006, pp.42, Washington, DC, USA, 2006.
[20] Strain, J.: Semi-Lagrange methods for level set equations, J Comput. Phys 151(2):498–533. DOI:10.1006/jcph.1999.6194
[21] Turk, G.: Re-Tiling polygonal surfaces, Computer Graphics(SIGGRAPH 92 Conference Proceedings), 26(2), pp. 55-64, July 1992.
[22] Wang, M.-Y.;Wang, X.-M.; Guo, D.-M.: A level set method for structural topology optimization, Comput. Methods Appl. Mech. Eng. 192(1-2), 2003, 227–246. DOI:10.1016/S0045-7825(02)00559-5
[23] Wang, L.; Li J,; Hagiwara, I.; A topology preserved mesh simplification algorithm, In Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering (CSIE '09), Vol. 1. IEEE Computer Society, Washington, DC, USA, 698-702. DOI:10.1109/CSIE.2009.405

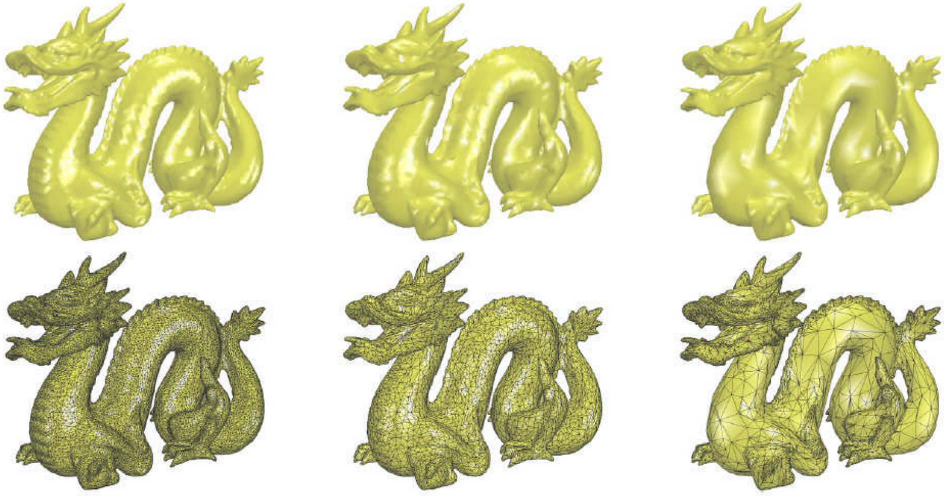

Fig. 6: Dragon, (left) original solid and mesh model; (middle) MeshLab; (right) our method.

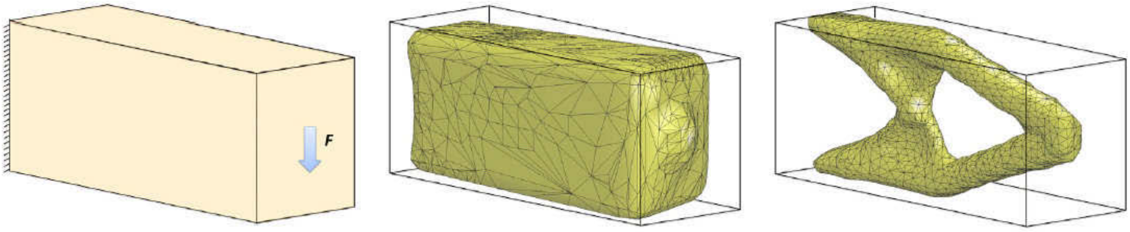

Fig. 7: Compliance minimization of a cantilever beam: (left) initial configuration with full material, load is applied at the middle of front right face, the back left face is constrained with zero displacement, upper volume is constrained to 10 percent of the origin; (middle) simplified model in intermediate step; (right) optimal design.