TOWARD STRESS-ALIGNED LATTICE DESIGN

By

Qiren Gao

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Mechanical Engineering—Master of Science

2021

ABSTRACT

TOWARD STRESS-ALIGNED LATTICE DESIGN

By

Qiren Gao

A novel approach to designing lattice structures is presented in which lattice members are aligned with the principal stress directions, and the lattice spacing and member sizing are optimized simultaneously to obtain an optimized design. By allowing both the lattice layout and the member sizes to vary during a multi-level design optimization process, this approach allows a greater flexibility in searching the design space and finding potentially higher performing designs than standard topology or triangularization methods. Example applications amply illustrate the capability of this approach to produce high quality designs.

TABLE OF CONTENTS

LIST OF FIGURES

KEY TO SYMBOLS AND ABBREVIATIONS

$\leq$	Less than or equal to
$[x]$	Matrix of x (or any contents within)
$\lvert x \rvert$	Determinant of x (or any contents within)
$\sum_{i=1}^{n} x_i$	Sum of all x_i where i varies from 1 to n
AM	Additive Manufacturing
TO	Topology Optimization
2D	2-Dimensional
3D	3-Dimentional
MMA	Method of Moving Asymptotes
FEM	Finite Element Method
SAL	Stress-aligned lattice
TLL	Triangular layout lattice
PWH-Axial	Plate with a hole under axial loading
PWH-Cantilever	Plate with a hole under bending (as a cantilever beam)
DOF	Degree of freedom

1. INTRODUCTION

Lattice structures are common in nature because they provide a light-weight but stiff structure. For example, the micro-structure of bone [1], animal protein structures [2], the structure in plant cells [3], and many other naturally occurring structures have lattice-type properties. Historically, manufacturing small-scale lattice structures was difficult, but additive manufacturing (AM) allows complex lattice structure designs to be produced at various and even at multiple scales. In addition to structural and material-focused applications, lattice structure designs are also well suited to multi-physics applications such as thermal [4]–[7], acoustic damping [8], [9], vibration and band gap [10], [11], magnetic fields [12] and more.

Design concepts that use periodic cellular lattices are the most common ways of designing a lattice structure [13]–[15]. For periodic cellular lattice structures, various studies have focused on optimizing the size of the unit lattice cell [16], [17]; mechanical properties for different unit lattice cells [18], [19]; and sandwich lattice structures [20], [21].

Recent studies have investigated combining topology optimization (TO) and lattice structure design. Some of these studies used the idea of periodic cellular lattice design, but the relative densities for the lattice cells were varied based on the results from topology optimization [22], [23]. The resulting structures are called graded lattice structures.

Daynes [24] investigated the concept of aligning the lattice members with the isostatic stress lines. The isostatic stress lines are also called the stress (or strain) trajectories and are related to the load paths [25], [26]. This concept has also been applied using strain trajectories [27].

Additional studies involving the alignment of lattice members to the stress (or strain) trajectories have been performed [28].

The current work introduces a novel approach to generating lattice structures in which every lattice member is aligned with one of the principal stress directions, and the lattice spacing and member sizing are optimized simultaneously using a bi-level optimization algorithm to obtain an optimal design. By allowing both the lattice layout and the member sizes to vary during a multi-level design optimization process, this approach allows for greater flexibility in the design space and potentially higher performing designs.

2. BI-LEVEL LATTICE OPTIMIZATION

The primary objective of this study is to generate the stiffest possible lattice structure using a prescribed amount of material. Both the lattice layout (spacing and orientation) and the cross-sectional radius of each lattice member play a role in achieving this goal [24]. Moreover, these two sets of design properties are strongly coupled. The optimal set of lattice member radii will be different for each lattice layout, which makes it challenging to search on both sets of properties at the same time. For these reasons, a bi-level optimization strategy [29], which is gaining increasing attention in the optimization field, was used in this study.

2.1. ASSUMPTIONS AND FUNDAMENTAL DESIGN CRITERIA

This work was inspired by the previous study [16], which is about finding the optimal sizing for a unit lattice cell that form the infill as a replacement of the solid material with different design domains and boundary conditions. The major study focus for the stress-aligned lattice design is to optimize the orientation and sizing for every single lattice member to its most efficient status in terms of stiffness.

Several assumptions are made regarding the lattice model:

- Lattice members are considered to be straight beams with a uniform and circular cross-section, and the cross-sectional radius for different members can vary,
- Buckling for beams under compression is ignored,
- Lattice members are connecting to others at a point solid joint, in other words, the entire lattice structure is considered to be a frame structure,
- The material is assumed to be homogeneous and isotropic.

The primary design goal for this work is to align the axial direction of each lattice member to a local principal stress direction, which is obtained by analyzing a model that treats the entire design domain as a uniform solid model. Given the above assumptions, a lattice member may experience tension, compression, torsion, bending, and shear. Aligning a lattice member to the local principal stress direction efficiently reduces the shear stress. Such design criteria also allow a lattice member to be loaded mainly under normal stress, which is an ideal loading situation for a beam structure.

2.2. BI-LEVEL LATTICE DESIGN STRATEGY

The global objective for this bi-level optimization study is to minimize the structural compliance (equivalent to maximizing the stiffness) of the structure. Parameters that control the lattice layout will be optimized in the 1^{st} (upper) level optimization based on the design domain geometry and the stress field for a uniform material. The 2^{nd} (lower) level optimization will only optimize the radius of each lattice member based on a given lattice layout structure provided by the upper level.

The problem statement for the bi-level lattice optimization study is stated as:

1st (upper) level optimization:

Objective:

- minimize the compliance of the lattice structure

Constraints:

- No specific constraints other than basic constraints like remaining the design domain unchanged

Variables:

- x and y coordinate of the start point
- maximum and minimum allowable distance between the seed points: $MaxGap$, $MinGap$
- radius of the merge circle: R
- spacing constant: α

2nd (lower) level optimization:

Objective:

- minimize the compliance of the lattice structure

Constraints:

- total volume of the lattice structure is less than a prescribed value

Variables:

- radii of lattice members: r_i, s.t. $r_i^{min} \leq r_i \leq r_i^{max}$

  for $i = 1, 2, \ldots total\ number\ of\ lattice\ members$

  where r_i^{min} and r_i^{max} represent the minimum and maximum allowable radius for the i^{th} lattice member

A flowchart that describes the workflow of the bi-level lattice structure optimization study is shown in Figure 1.

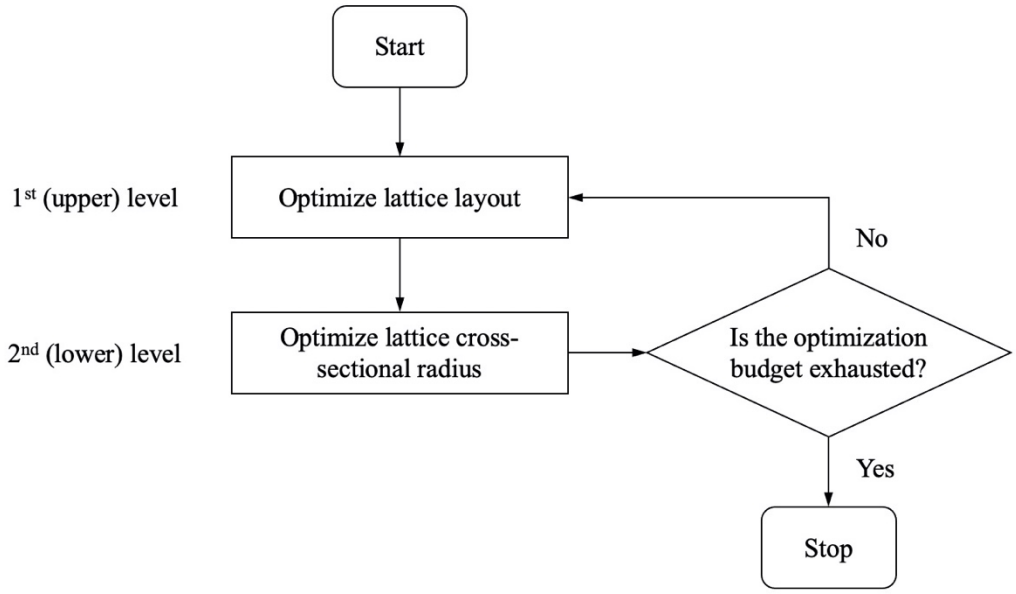


Figure 1. Flowchart of the bi-level lattice structure optimization study.

In this bi-level lattice structure optimization study, the multi-disciplinary design exploration tool HEEDS MDO [30] is used to automate the entire process and to perform the first- (upper-) level optimization. The hybrid adaptive design exploration method SHERPA is used to optimize the layout parameters in the first- (upper-) level. The lattice layout generation algorithm was developed in the Wolfram Mathematica language [31], [32]. This algorithm takes data vector fields (a stress field in this study), layout parameters, and the design domain as inputs. It then generates the nodal coordinates, connectivity, and the upper bound and lower bound for the radius of each lattice member. The flowchart in Figure 2 shown the details regarding the bi-level optimization. The steps marked within the dashed purple box are the procedures for the first- (upper-) level optimization while the steps marked within the solid green box are the procedures for the second- (lower-) level optimization.

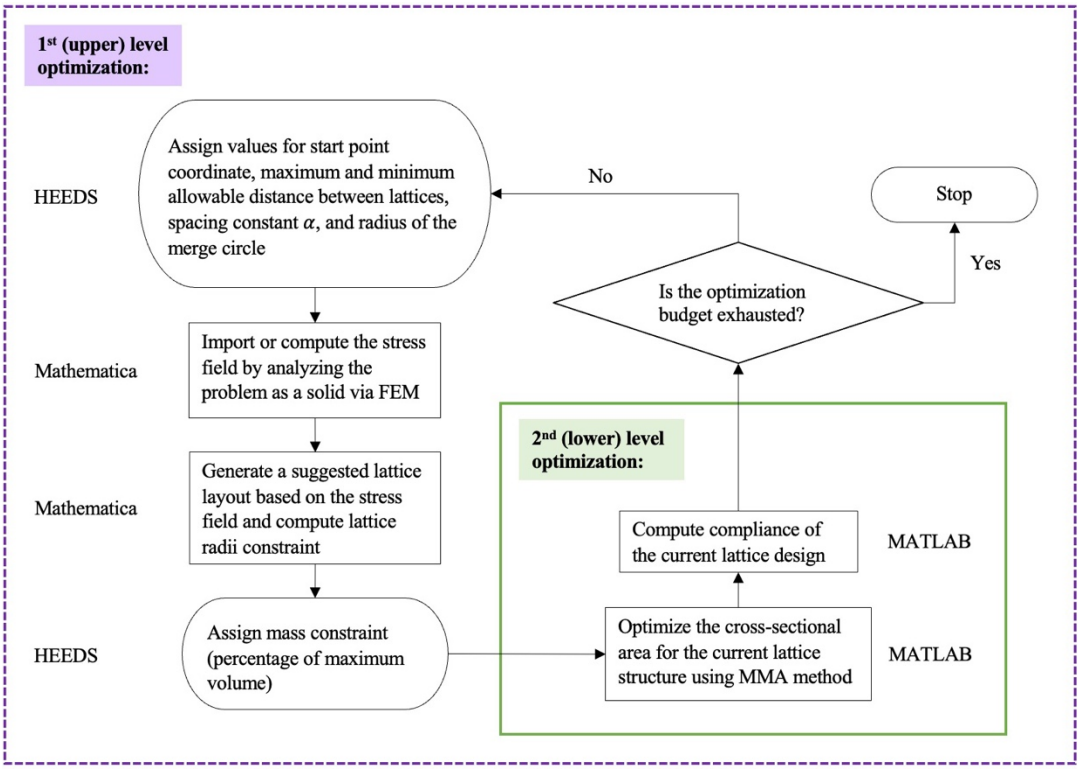


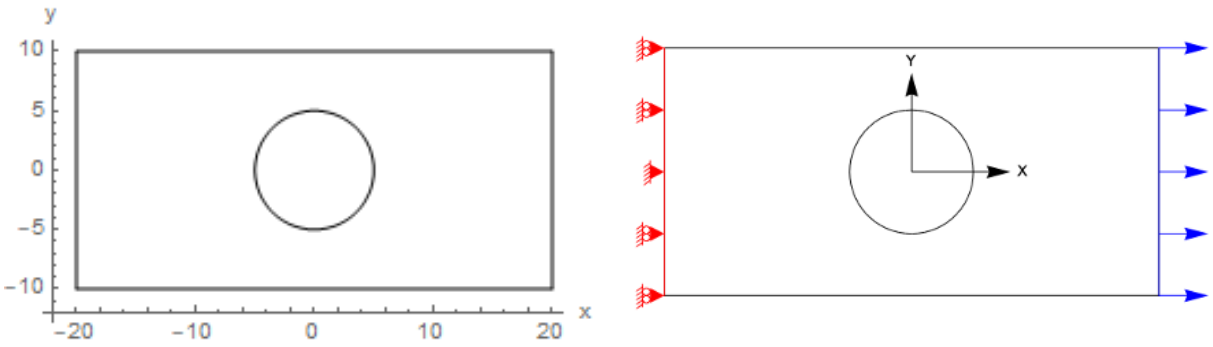
Figure 2. A flowchart of the bi-level lattice structure optimization study with more details. The steps marked within the solid green box belong to the 2nd (lower) level optimization, while the steps marked within the dashed purple box belong to the 1st (upper) level optimization. The software used for each step is labeled beside the step block.

Within the *second*- (lower-) level optimization, MATLAB [33] takes the output from the lattice layout generation algorithm as input and uses the Method of Moving Asymptotes (MMA) search algorithm [34] to optimize the radius of each lattice member. The lattice structure needs to be re-analyzed at each iteration for computing the compliance and the solution gradients, which are used within MMA. Each lattice member is assumed to be a circular cylinder.

3. LATTICE LAYOUT GENERATION

3.1. STRESS FIELD GENERATION

To illustrate the stress-field-aligned lattice generation process, we will use the example problem shown in Figure 3. The example problem is defined as a $40\ mm \times 20\ mm$ rectangular plate that has a $5\ mm$ radius hole at its center. The plate thickness is $2\ mm$. The left boundary is fixed against horizontal (x-direction) motion at all points, while the mid-point of this boundary is also fixed against vertical motion (y-direction). The right boundary is uniformly loaded with a positive x-direction stress. This problem is a finite version of the infinite plate with a hole problem studied by Kirsch [35], [36].



(a) (b)

Figure 3. The design domain for the example problem is a solid 2D plate with a circular hole in the center. The left boundary is fixed in the x-direction. The right boundary is loaded with a uniform stress in the x-direction.

The original infinite plate with a hole problem stated in [36] was about studying a linear elastic solution for stresses around a hole when the plate is under uniaxial tension. According to the results from [36], the local principal stress value of the region on a line that is perpendicular to the uniaxial tension can be summarized as in Figure 4. The infinite long plate with a hole that has

a radius of R is loaded with uniaxial tension σ_∞ as shown in this figure. The principal stress value at the location along the represented line where r is equal to R is 3 times the uniaxial tension loading, while the stress value decreases exponentially till it reaches the same value of the uniaxial tension σ_∞ when the value of r is increasing. This well-studied problem gives a good intuition to the behavior of the stress field while it has a proper design difficulty given by the hole in the middle of the design domain.

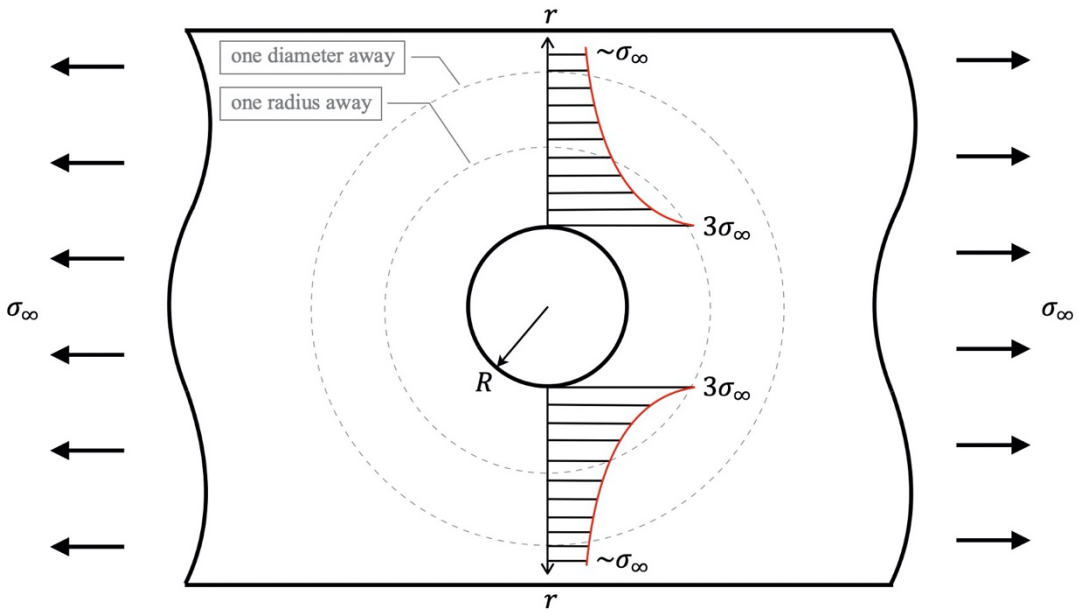
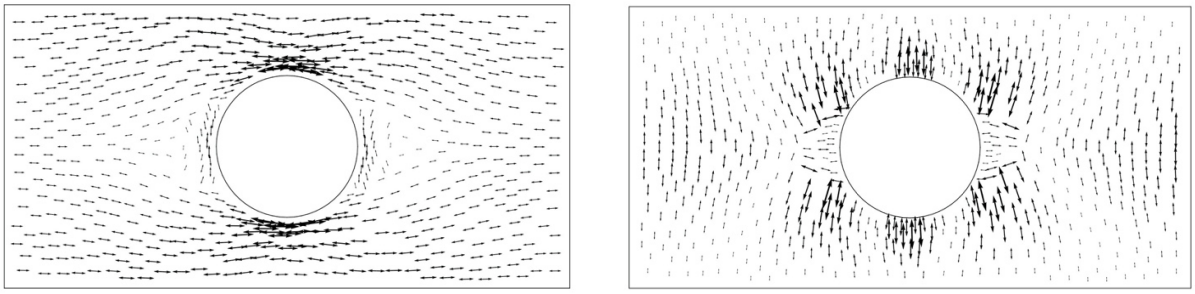


Figure 4. The stress state of the region around the hole in the Kirsch problem. The curve of stress (highlighted in red) represents the principal stress value along the vertical line that is perpendicular to the uniaxial tension.

An estimate of the stress distribution within the design domain is established by analyzing the problem as a continuum (e.g., plane stress in 2D) [32] using the finite element method (FEM) [37]. The material is assumed to be isotropic and homogeneously distributed within the domain. For this work, the analysis for the solid models was processed by the built-in FEA package in Mathematica. Locally, the most efficient orientation of an orthotropic material is one in which its

principal material directions coincide with the principal stress directions. Similarly, it assumed here that optimally oriented lattice members will be aligned with the principal stress field directions obtained from the continuum analysis. This corresponds to an interpretation that the lattice local behavior is similar to that of an orthotropic material [19], [38], [39]. For the current example problem, Figure 5 shows the first and second principal stress fields corresponding to the given geometry and loading of the design domain, analyzed as a continuum. The size of the arrows is proportional to the local corresponding principal stress values, which are computed at the middle of each two-headed arrow.



(a) (b)

Figure 5. (a) The first and (b) second principal stress fields in the design domain of the example problem.

For a two-dimensional problem, the stress matrix in Equation 1 yields two eigenvalues, which correspond to the first and second principal stress while the two eigenvectors correspond to the first and second principal stress directions. The two principal stress vector fields can be generated from these eigenvectors, as shown in Figure 5 (a) and (b) for the current example. At any location, the two eigenvectors (local principal stress directions) of the following stress matrix are perpendicular to one other.

$$\sigma_{(x,y)} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{bmatrix} \qquad \textit{Equation 1}$$

A sharp direction change of the principal stress direction can be observed from Figure 5(a) and Figure 5(b). This is a numerical computation issue that happened because the stress distribution at any location in those regions is 'flat', which says, the value of 1^{st} and 2^{nd} principal stresses in those regions are very close. The method that been used in this work to ensure the stress trajectories are generated correctly through these regions was stated in Section 3.2.

Here, we aim to align the lattice members to the local stress field. Note that it is possible to extend this approach to generate lattices that are aligned to other fields related to, for example, heat transfer [6], electromagnetic [40], acoustics [8], and others.

3.2. GENERATING STRESS TRAJECTORIES

To generate lattices that are aligned to the stress field, we generate stress trajectories [25] from the principal stress field and convert the trajectories into a lattice layout. Such trajectories can be built by interpolating the principal stress directions throughout the design domain.

The process of stress trajectory generation begins by selecting an arbitrary point, referred to here as the start point. Two perpendicular stress trajectories will pass through this point. In each direction, the stress trajectory is generated by tracing the eigenvectors using Euler's method [26], [41] with a very small step size. The step size can be varied based on the needed accuracy. For the current example problem, the step size is set to $0.1\ mm$. A similar stress trajectory generation method is presented in [42]; the main difference lies in the stress trajectory tracing direction computation method, as discussed below. The principal stress trajectories generated from a start

point are called the start lines. Figure 6 shows an example of two start lines passing through an arbitrary start point (x, y).

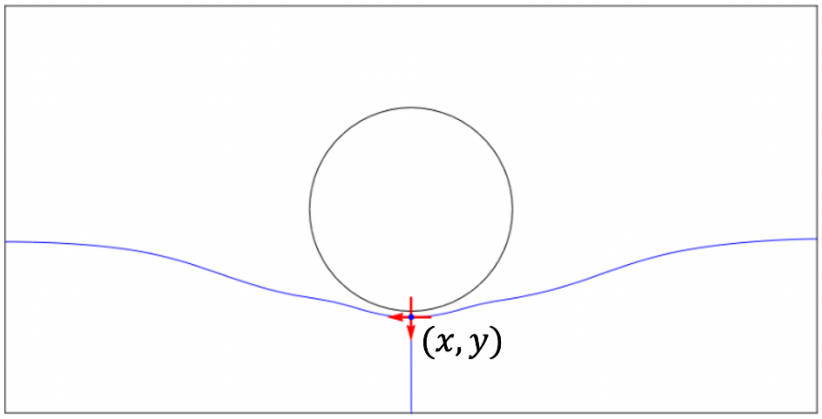
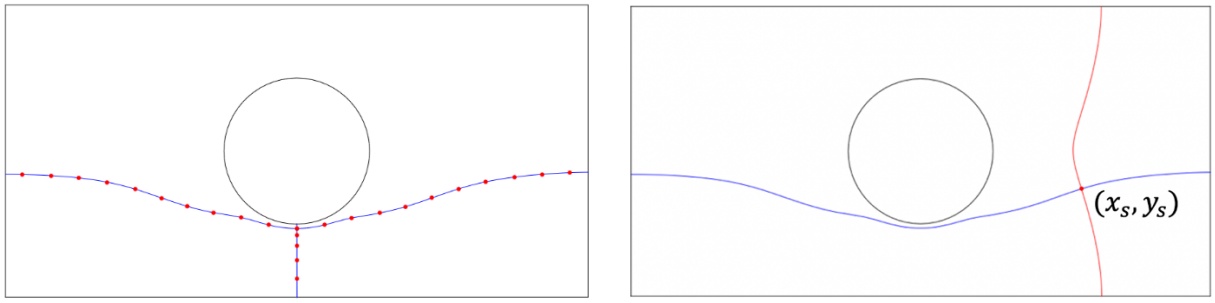


Figure 6. Two start lines generated from an arbitrary start point (x, y).

The issue of a sharp change of principal stress direction was stated in Section 3.1. To resolve such an issue, the stress trajectory generating algorithm involved using a direction check process. As described above, a stress trajectory is generated with many small steps. The direction check takes the direction vector generated from the previous step as contrast and chooses the closest stress direction vector computed from the current location as the 'direction to go' for this step.

Seed points are generated along a start line, at a spacing that is described below. At each seed point, a new stress trajectory is generated that passes through and is perpendicular to the start line. Figure 7(a) shows the seed points generated on both start lines, and Figure 7(b) shows an example of a stress trajectory that is generated from a seed point.



(a) (b)

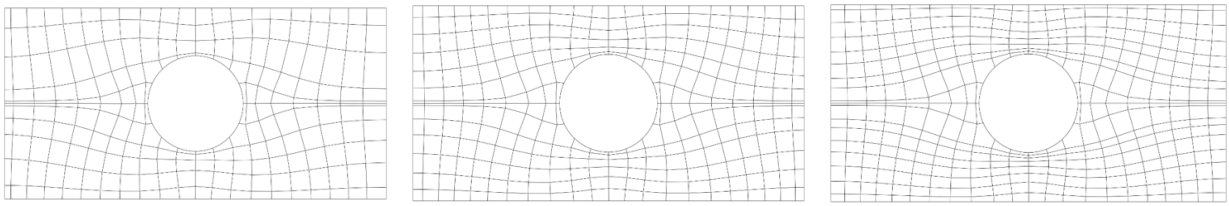
Figure 7. (a) Seed points are generated along both start lines. The distance between seed points is computed based on local stress values. (b) A sample stress trajectory passing through a seed point (x_s, y_s).

In addition to the lattice direction, the relative distance between parallel lattice members is determined by the local stress value. The maximum and minimum allowable distances between lattice members are assigned at the beginning of the generation process. These values are specified by the user's intuition or, here, controlled by the optimization software HEEDS MDO [30], which is described in Section 2.2. The distance between lattice members is then inversely related to the magnitude of the local stress values. The distance between seed points is thus determined using the following relation:

$$\frac{d - d_{min}}{d_{max} - d_{min}} = \left(\frac{\sigma_{(x_s,y_s)} - \sigma_{global\ min}}{\sigma_{global\ max} - \sigma_{global\ min}} \right)^{\alpha} \qquad \textit{Equation 2}$$

where d is the distance to the next seed point from the current seed point at (x_s, y_s), d_{min} and d_{max} are the user-defined minimum and maximum allowable distances between lattice members, $\sigma_{(x_s,y_s)}$ is the local principal stress value of the seed point at (x_s, y_s), $\sigma_{global\ min}$ and $\sigma_{global\ max}$ are the global minimum and maximum principal stress values, respectively, in the entire design domain, α is a spacing constant that determines how sensitive the distance d is relative to the

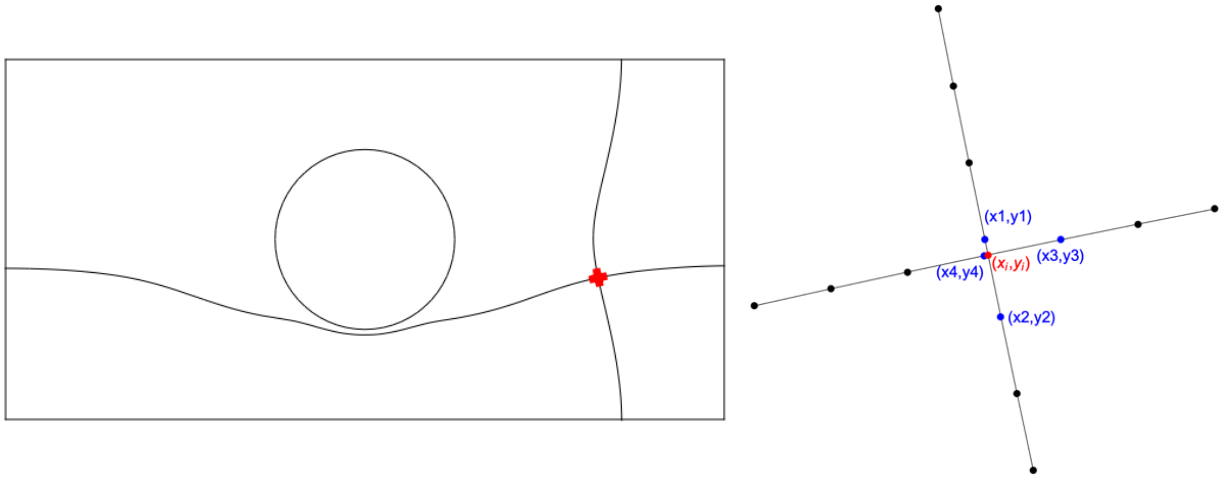
local stress value $\sigma_{(x_s,y_s)}$. The spacing constant α is controlled by the user or by the optimization algorithm. Examples of how the spacing constant α affects the lattice layout are shown in Figure 8. The number of the stress trajectories is proportional to the spacing constant α value. We choose $\alpha = 1$ for the example problems in this paper.



(a) (b) (c)

Figure 8. Examples of how the spacing constant α affects the lattice layout where (a) $\alpha = 0.5$*, (b)* $\alpha = 1.0$*, (c)* $\alpha = 1.5$*.*

By treating each stress trajectory or boundary line as a piece-wise straight line, we can compute the intersections that form between two stress trajectories or between a stress trajectory and a boundary. The method used here is illustrated in Figure 9. The intersection of the two lines can be computed using equation (3), using concepts related to those found in [43].



(a) (b)

Figure 9. (a) Two stress trajectories forming an intersection in the highlighted section; (b) A zoomed in view of the highlighted sections of both stress trajectories shown in (a), point (x_i, y_i) *is the intersection.*

$$x_i = \frac{\begin{vmatrix} \begin{vmatrix} x1 & y1 \\ x2 & y2 \end{vmatrix} & \begin{vmatrix} x1 & 1 \\ x2 & 1 \end{vmatrix} \\ \begin{vmatrix} x3 & y3 \\ x4 & y4 \end{vmatrix} & \begin{vmatrix} x3 & 1 \\ x4 & 1 \end{vmatrix} \end{vmatrix}}{\begin{vmatrix} \begin{vmatrix} x1 & 1 \\ x2 & 1 \end{vmatrix} & \begin{vmatrix} y1 & 1 \\ y2 & 1 \end{vmatrix} \\ \begin{vmatrix} x3 & 1 \\ x4 & 1 \end{vmatrix} & \begin{vmatrix} y3 & 1 \\ y4 & 1 \end{vmatrix} \end{vmatrix}}, \quad y_i = \frac{\begin{vmatrix} \begin{vmatrix} x1 & y1 \\ x2 & y2 \end{vmatrix} & \begin{vmatrix} y1 & 1 \\ y2 & 1 \end{vmatrix} \\ \begin{vmatrix} x3 & y3 \\ x4 & y4 \end{vmatrix} & \begin{vmatrix} y3 & 1 \\ y4 & 1 \end{vmatrix} \end{vmatrix}}{\begin{vmatrix} \begin{vmatrix} x1 & 1 \\ x2 & 1 \end{vmatrix} & \begin{vmatrix} y1 & 1 \\ y2 & 1 \end{vmatrix} \\ \begin{vmatrix} x3 & 1 \\ x4 & 1 \end{vmatrix} & \begin{vmatrix} y3 & 1 \\ y4 & 1 \end{vmatrix} \end{vmatrix}} \qquad \textit{Equation 3}$$

where $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$ denotes the determinant of the matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$.

With the collection of stress trajectories and their intersections, we obtain the result shown in Figure 10.

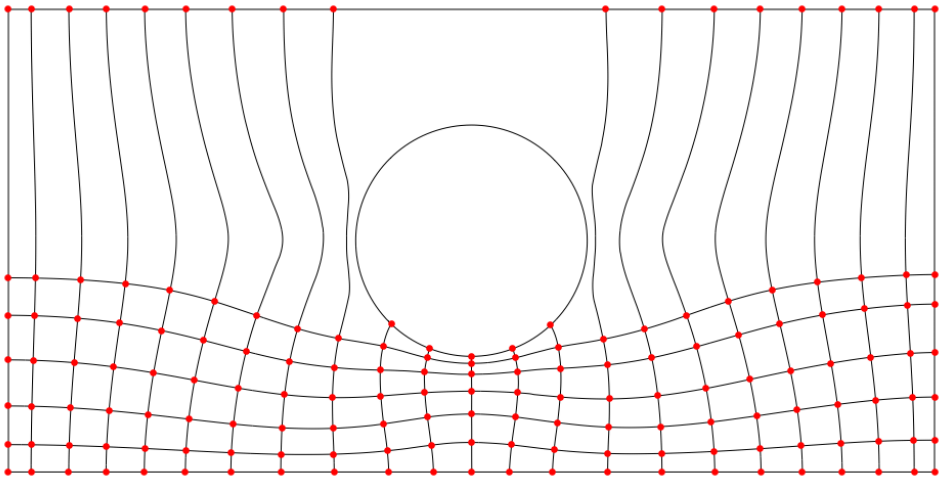
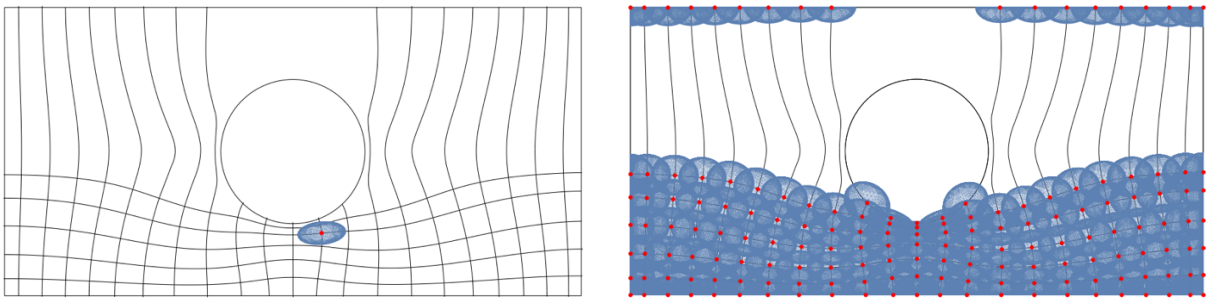


Figure 10. Stress trajectories generated by seed points and the intersections of all trajectories and boundaries, all based on a single start point.

3.3. DEFINE THE UNFILLED REGION

The initial set of trajectories shown in Figure 10 do not fill the entire design domain. We identify the unfilled regions by subtracting elliptical regions at each trajectory intersection from the design domain. The size of each ellipse is related to the local principal stress values, which also determine the spacing of the trajectories. Figure 11(a) shows an example of one ellipse that is generated at an intersection. In Figure 11(b), the region that is not covered by any ellipse is considered as the unfilled region.



(a) (b)

Figure 11. (a) Example of one stress-related ellipse region; (b) all stress-related ellipse region generated from intersections.

The ellipses are defined as follows. Each trajectory intersection serves as the center of an ellipse. The semi-minor axis and the semi-major axis of each stress-related ellipse are computed using Equation 4 and Equation 5, respectively.

$$SMinor = Tol\frac{\sigma_{global\ max} - \sigma_{2nd}^{int}}{\sigma_{global\ max} - \sigma_{global\ min}}(d_{max} - d_{\min}) + d_{min} \qquad \textit{Equation 4}$$

$$SMajor = Tol\frac{\sigma_{global\ max} - \sigma_{1st}^{int}}{\sigma_{global\ max} - \sigma_{global\ min}}(d_{max} - d_{\min}) + d_{min} \qquad \textit{Equation 5}$$

where $SMinor$ and $SMajor$ are the semi-minor axis and the semi-major axis of the ellipse, Tol is a user-defined tolerance, $\sigma_{global\ min}$ and $\sigma_{global\ max}$ are the global minimum and maximum principal stress values, d_{min} and d_{max} are the minimum and maximum allowable distances between lattice members, σ_{1st}^{int} and σ_{2nd}^{int} are the 1^{st} and 2^{nd} principal stress value computed at the intersection. The tolerance, Tol, is a positive constant that can be varied to allow more ($Tol > 1$) space or less ($Tol < 1$) space for the subsequent process of generating additional

stress trajectories. The default value for the variable Tol is 1. Each ellipse is rotated to align with the first eigenvector of the local stress matrix that is computed at the intersection.

The ellipse region is smaller when the local principal stress values are closer to the global maximum value to gives more room for design. As stated in Section 3.1, the region below the circle has a larger stress value compared to the region that is further from the circle. The shape of the ellipse regions shown in Figure 12 follows the stress state: the ellipses are 'flatter' in the middle region where the 1^{st} principal stress value is high but the 2^{nd} principal stress value is low; the shape of the ellipses are generally larger and closer to a circle when they are further from the hole.

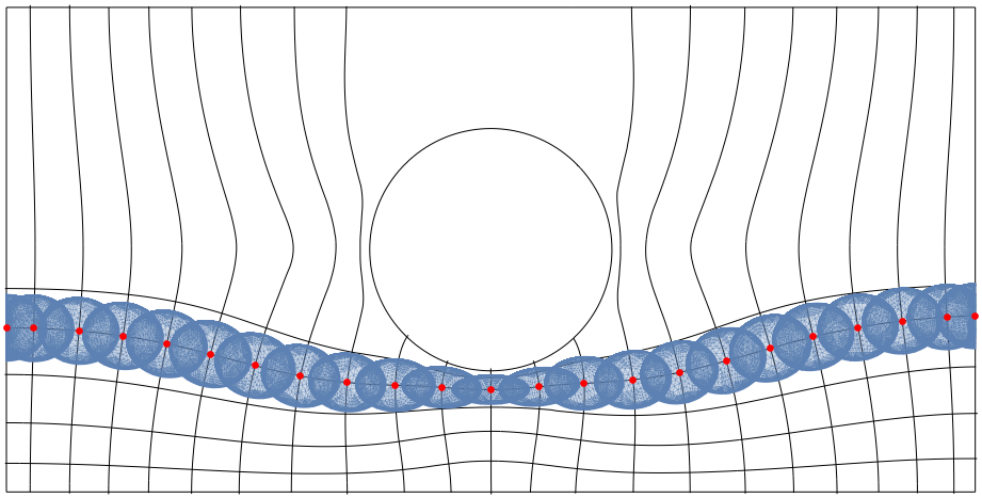


Figure 12. Examples of the stress-related ellipse region generated on points that has different local principal stress values.

3.4. RANDOM START METHOD

In this work, two methods have been investigated for filling the unfilled regions with lattice members: random start and crystal growing. Both are iterative processes. An iteration for both

methods starts by computing intersections of all lines (including stress trajectories and boundary lines) to determine the unfilled region, and it ends when either no new stress trajectories are generated or when it reaches the maximum iteration number. The definition of the maximum iteration number depends on the method.

The random start algorithm for filling the unfilled regions with lattice members follows these steps:

Step 1. Compute the unfilled region as described in the previous section.

Step 2. Randomly generate a user-defined number of start points in the entire design domain

Step 3. Eliminate start points in the already-filled region. If no new start points remain, then end the process.

Step 4. If the number of allowable iterations is exceeded, then end the process.

Step 5. Randomly choose a start point from the group of candidate start points to generate start lines and seed points, as described in Section 3.2.

Step 6. Filter the seed points as described below.

Step 7. If no seed points remain after filtering, return to Step 4.

Step 8. Generate new stress trajectories from the remaining seed points.

Step 9. Return to Step 1

The user defines the number of new start points to be randomly generated within the design domain. After filtering out the start points located within any ellipse at the existing trajectory intersections, the remaining start points are called the candidate start points (Figure 13). The number of candidate start points defines the maximum allowed iterations. The definitions of start point, start line, and seed point are the same as described in Section 3.2.

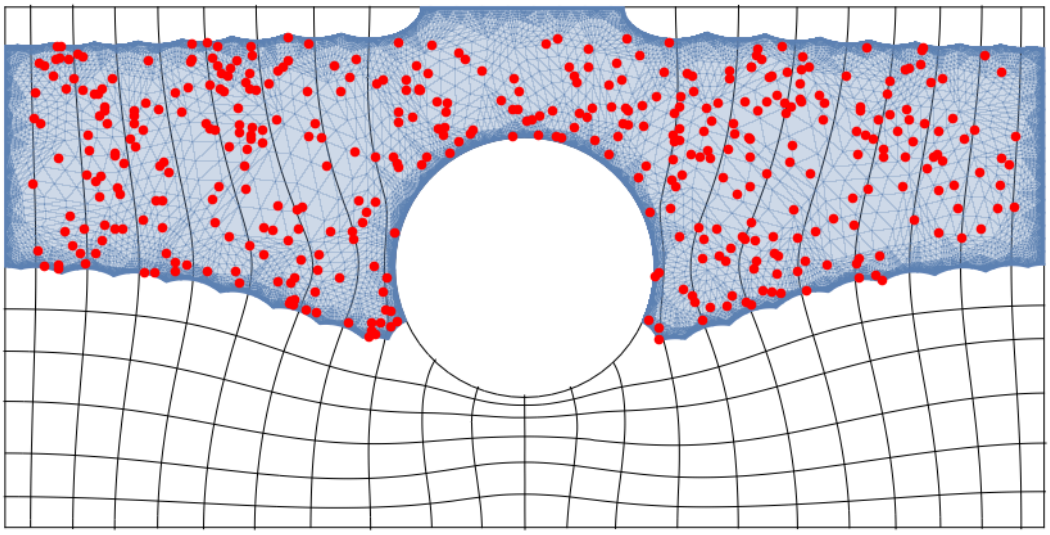
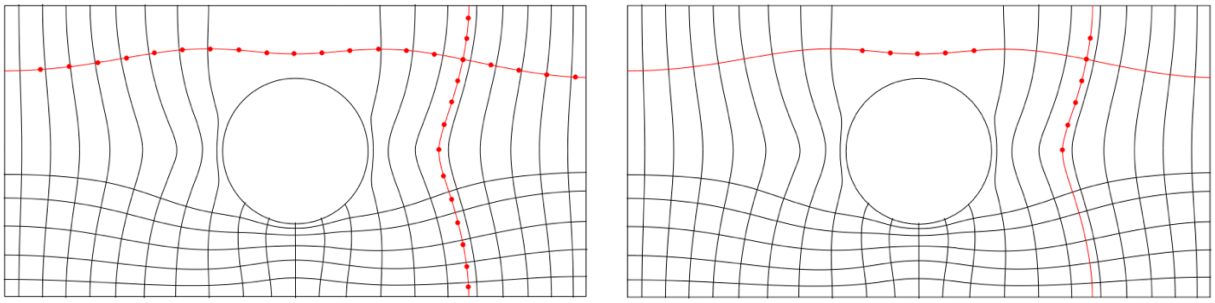


Figure 13. The highlighted blue region represents the unfilled region. Red points are the new start points in the unfilled region.

During each iteration, one of the new start points is randomly chosen. New start lines (stress trajectories) are then developed as described previously, by creating two orthogonal trajectories and generating seed points on these start lines, as in Figure 14(a). Seed points are deleted if they are located outside the unfilled region or if they are in close proximity to any previously generated trajectories that are "perpendicular" to the start line on which they were generated. Figure 14(b) illustrates the result of filtering seed points that were generated in Figure 14(a). Stress trajectories will be generated from the remaining seed points.



(a) (b)

Figure 14. (a) Example of how new seed points are generated and filtered. The red lines represent the start lines. Red points represent the seed points. (b) The remaining seed points after filtering.

New stress trajectories are generated from the remaining seed points, which completes the current iteration. However, if none of the seed points are kept during the filtering process, then the process is repeated for another start point until at least one new trajectory has been generated or the maximum iteration number is reached.

3.5. CRYSTAL GROWING METHOD

The crystal growing algorithm for filling the unfilled regions with lattice members follows these steps:

Step 1. Compute the unfilled region as described in the previous section.

Step 2. Identify the existing stress trajectories that cross the unfilled region. These trajectories are called the candidate start lines. If no stress trajectories meet this criterion, then end the process.

Step 3. If the number of allowable iterations is exceeded, then end the process.

Step 4. Randomly choose a trajectory from the group of candidate start lines to generate seed points.

Step 5. Filter the seed points as described below.

Step 6. If no seed points remain after filtering, return to Step 4.

Step 7. Generate new stress trajectories from the remaining seed points.

Step 8. Return to Step 1

This method starts by counting the number of existing stress trajectories that are crossing the unfilled region. These lines are shown in red in Figure 15 (all the vertical trajectories in this case). The number of highlighted lines is the maximum number of iterations in the current process. The boundary lines are not included as candidate start lines.

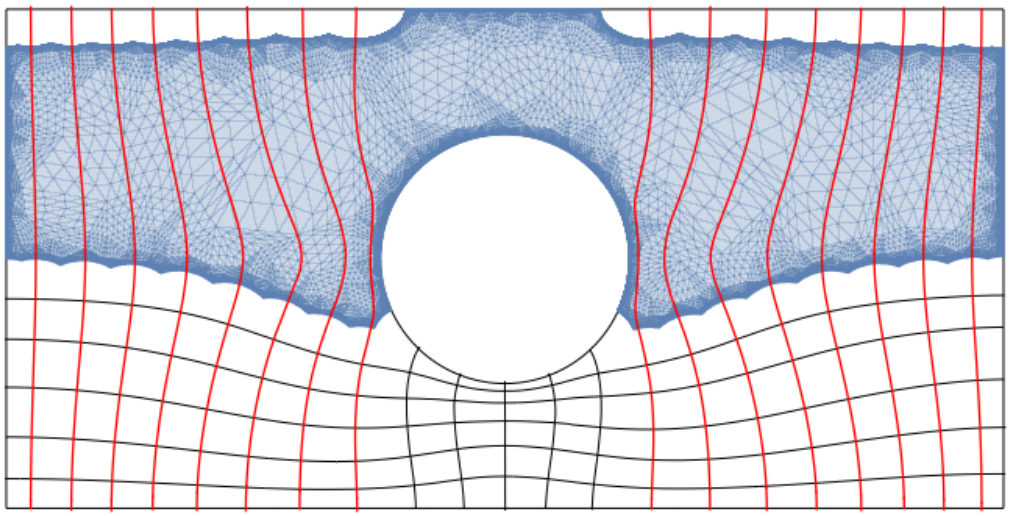


Figure 15. The highlighted blue region is the unfilled region. The red lines (vertical trajectories, in this example) are the existing trajectories that cross the unfilled region.

One of the existing trajectories is selected randomly, and new seed points are generated on it. The new seed points are then filtered, following the same criteria as in the random start method. New stress trajectories are generated from the kept seed points. The current iteration ends when

the new stress trajectories are generated. However, if none of the seed points are kept during the filtering process, then the process is repeated for another existing trajectory.

3.6. RESULT OF TRAJECTORY FILLING

Figure 16 shows a final result from the random start method. Because both methods involve randomized points or factors, different results may be obtained during each run. The random start method tends to be more computationally intensive compared to the crystal growing method. However, because the crystal growing method depends strongly on the first trajectory layout, it may not be as reliable in adequately covering the entire design domain with trajectories.

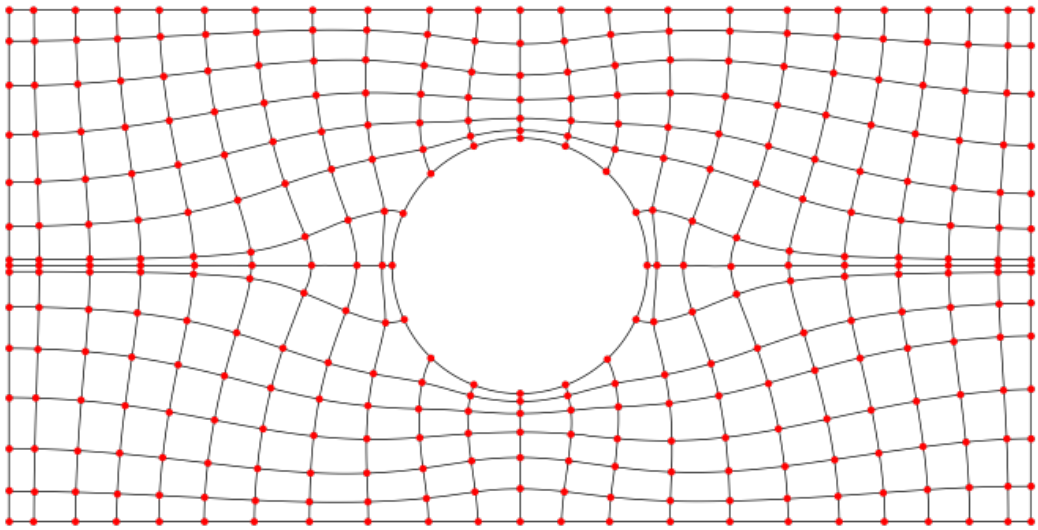


Figure 16. The design domain is completely covered by stress trajectories. Red points represent the line intersections.

3.7. CONVERTING STRESS TRAJECTORIES INTO A LATTICE

Stress trajectories serve as a basis for lattice member orientations. Once the design domain is completely covered by stress trajectories and the intersections between trajectories and

boundaries have been computed, the stress trajectories can be converted to a lattice. In this study, each intersection point is connected to adjacent intersection points by a straight line, representing a single straight lattice member. Figure 17 shows the result of this conversion. It is possible to represent the span between intersection points with multiple and/or curved lattice beam members, but these higher resolution models were not investigated here.

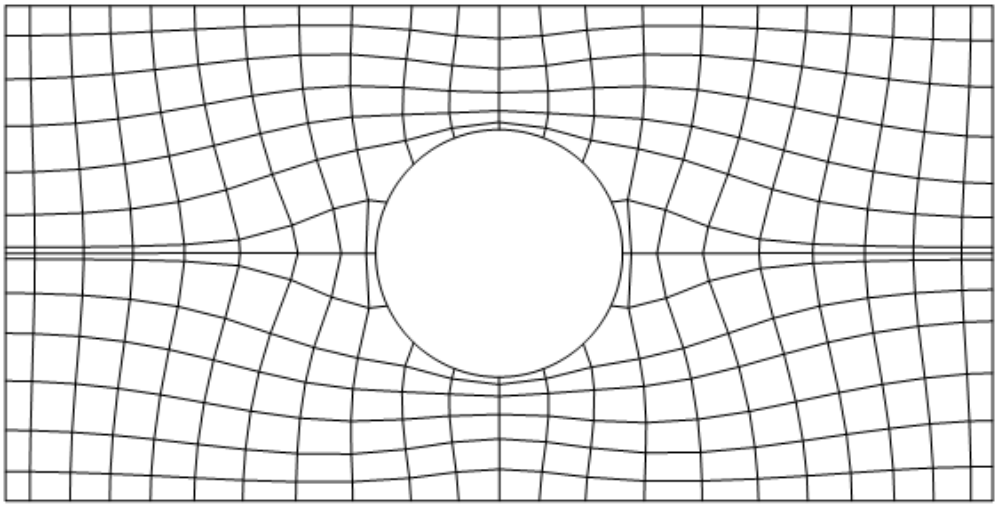
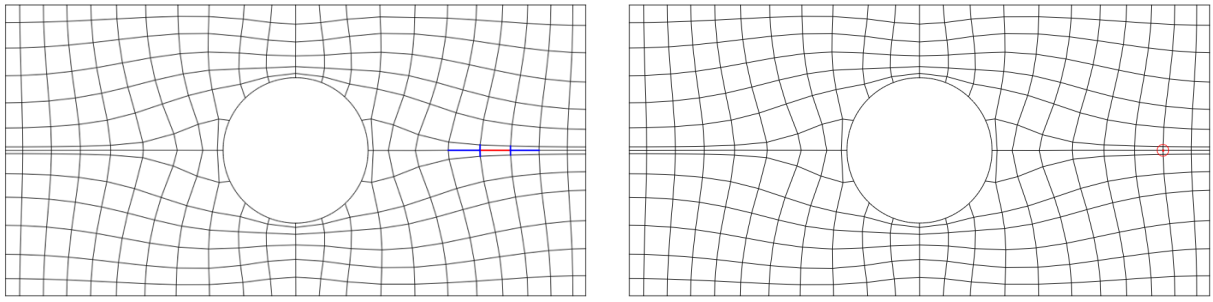


Figure 17. The suggested lattice layout from the stress trajectories. A single, straight lattice beam member connects the trajectory intersection points.

3.8. MERGING ADJACENT LATTICE MEMBERS

The lattice layout obtained from the previous step may contain some members that are closer to one another than is desirable. This often occurs in regions of high strain energy, but it can happen in other regions due to the randomness of seed point selection. Close spacing of lattice members may overly restrict the allowable radius of some members (see Figure 18(a)), which may be undesirable in some applications (e.g., buckling). It may also produce manufacturing

issues due to close spacing or small feature size. Therefore, a healing step is introduced to merge lattice members that are too close.



(a) (b)

Figure 18. (a) The highlighted lattices in blue are the neighbors of the lattice in red. (b) The red circle represents the merge circle, which contains multiple nodes.

To identify lattice members that are too close, a merge circle is generated at each lattice node (intersection), with the node as its center. The radius of the circle is either specified by the user or treated as a design variable during the optimization process described below. Since the lattices tend to be closer to one another in regions of high strain energy, the merging process described below is carried out in order from the node that has the highest local stress value to the node that has the lowest value.

All lattice nodes within a merge circle will be merged either to a newly computed point or to an existing node, based on the following priority criteria. Nodes that define the design domain boundary are given highest priority, and these node locations always remain unchanged, though other nodes may be merged with them. In the current example, the highest priority nodes include the 4 corner nodes that define the rectangular outer boundary shape and the original nodes that define the circle cutout. These nodes with the top merge priority remain unchanged so that the shape of the design domain will not be changed. All other nodes on the design domain

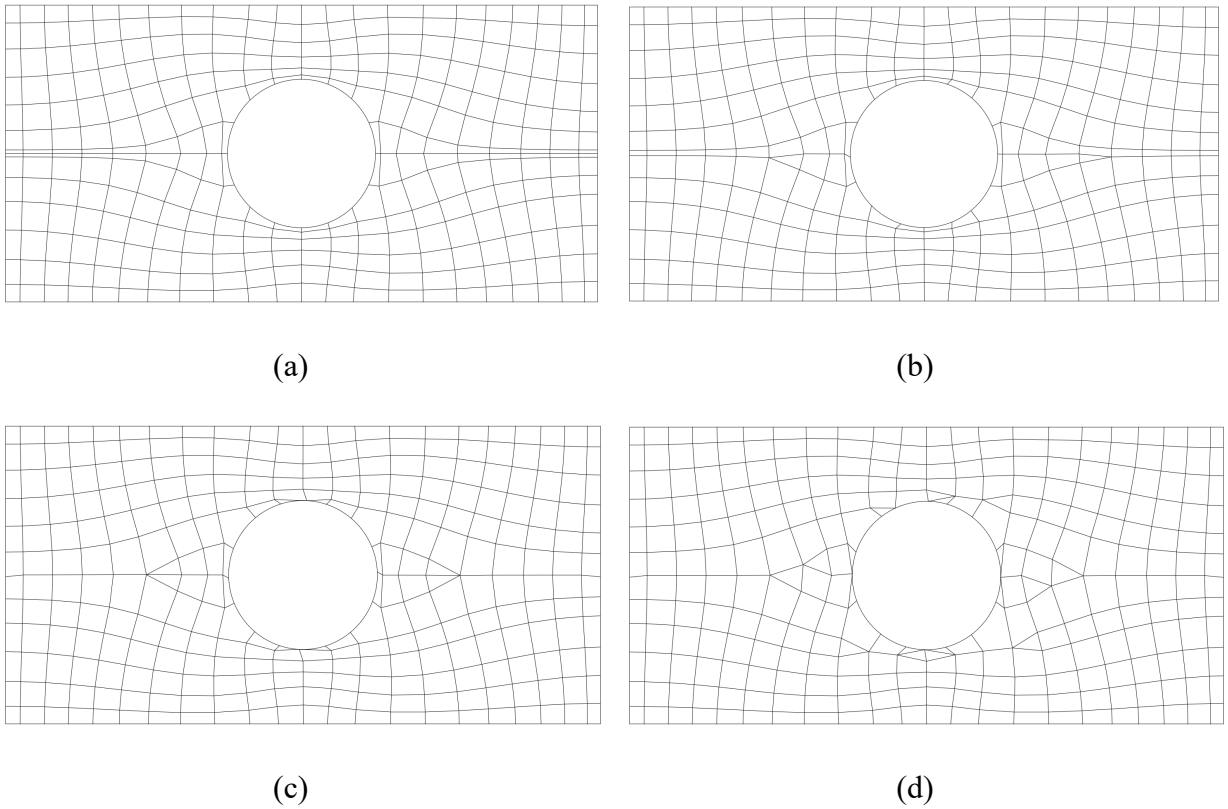
boundaries are assigned medium merge priority, and all interior lattice nodes are considered to be low merge priority nodes. Within a merge circle, nodes with lower priority will be relocated to the position of a node with higher merge priority. If all nodes share the same merge priority level, they are merged to a new node whose coordinates are a weighted average of the nodes being merged. The weighting is based on the local stress, as shown in Equation 6 and Equation 7:

$$coor_{new} = \sum_{i=1}^{n} w_i \times coor_i \quad where\ i = 1,2,\dots,n \qquad \textit{Equation 6}$$

$$w_i = \frac{\sigma_{coor_i}/\sigma_{global\ max}}{\sum_{i=1}^{n} \sigma_{coor_i}/\sigma_{global\ max}} \qquad \textit{Equation 7}$$

where $coor_{new}$ is the coordinate set of the new node, n is the number of nodes inside the merge circle, w_i is the weight of the i^{th} node in the merge circle, $coor_i$ is the coordinate set of the i^{th} node, σ_{coor_i} is the local stress value of the i^{th} node, and $\sigma_{global\ max}$ is the maximum stress value in the entire design domain.

Figure 19 shows the results of merging the example lattice using different sizes of merge circles. The reason for skipping the result of applying merge circle with a radius of 0.5 mm is because this result has little difference compared to the result given by using 0.4 mm as the radius of the merge circle due to the spacing between lattice for the current lattice layout.



(a) (b)

(c) (d)

Figure 19. Applying the lattice merging technique to the original lattice (a) produces the new layout lattice layouts. In (b) the radius of the merge circle is $0.3\ mm$*, in (c) the radius of the merge circle is* $0.4\ mm$*, and in (d) the radius of the merge circle is* $0.6\ mm$*.*

The result that is shown in Figure 19(c) has the best performance among the others. Therefore, we pick this result in this step and continue. In a general situation, the radius of the merge circle is considered to be an optimization variable within the bi-level optimization process.

4. LATTICE CROSS-SECTIONAL RADII OPTIMIZATION

With the lattice layout determined as described above, the cross-sectional radii for the lattice members are optimized using the MMA algorithm [34]. In this study, the MMA algorithm implementation within MATLAB [33] is used. As described in section 2.2, the lattice cross-sectional radii optimization is the second- (lower-) level optimization within the bi-level optimization process. The objective, constraints, and design variables are stated below:

2nd (lower) level optimization:

Objective:

- minimize the compliance of the lattice structure

Constraints:

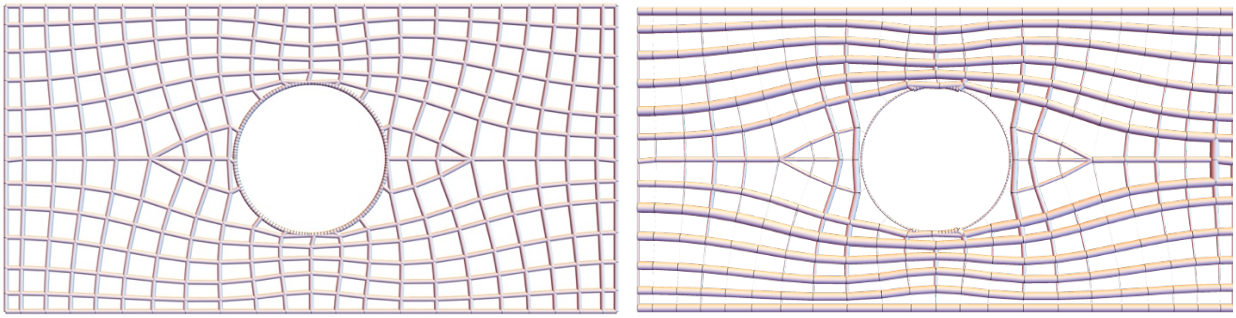
- total volume of the lattice structure is less than a prescribed value

Variables:

- radii of lattice members: r_i, s.t. $r_i^{min} \leq r_i \leq r_i^{max}$
  for $i = 1, 2, \ldots total\ number\ of\ lattice\ members$
  where r_i^{min} and r_i^{max} represent the minimum and maximum allowable radius for the i^{th} lattice member

Each lattice design candidate is analyzed using a linear static finite element model, also implemented within MATLAB. This model produces the nodal displacements, element strains and stresses, and the deformation gradients for each set of radii proposed by the optimization algorithm. The cross-sectional shape of each lattice member is assumed to be circular. A radius of $0.1\ mm$ was assigned as the cross-sectional radius for all lattice members to the 3D lattice

model before optimization (Figure 20(a)). For the layout shown in Figure 19(b), the sizing-optimized lattice structure for the current example is shown in Figure 20(b).



(a) (b)

Figure 20. (a) The 3D lattice model for the case when all lattice members have the same radius. (b) The optimized result for the lattice structure (Figure 19(b)) in the example problem.

For this example problem, the radii for the boundary lattice members have a lower bound of $0.1\ mm$ to ensure the boundary skin does not get eliminated. The need for a boundary skin layer is problem dependent. The radii for other interior lattice members have a lower bound of $0.001\ mm$, allowing these lattice members to effectively be eliminated during the optimization process while avoiding numerical errors. This minimum value for the radii can depend on the problem goals and/or the resolution of the manufacturing process.

4.1. COMPARISON TO A LATTICE OF ARBITRARY TRIANGULAR LAYOUT

Triangular layouts are commonly found in both 2D and 3D lattice structure design [44], [45]. A comparison between the proposed Stress-Aligned Lattice (SAL) model and an arbitrary Triangular Layout Lattice (TLL) model, as an application of the cross-sectional radii optimization process, will be conducted in this sub-section.

Figure 21 shows the lattice layout of a TLL model. The lattice layout for this TLL model is generated by a built-in linear triangle mesh generation algorithm in Mathematica, with a goal of having a number of lattice elements very similar to that in the SAL model. The resulting TLL model has 518 interior lattice elements compared to 516 interior lattice elements in the SAL model. Interior lattice elements are those that do not form the geometric boundaries.

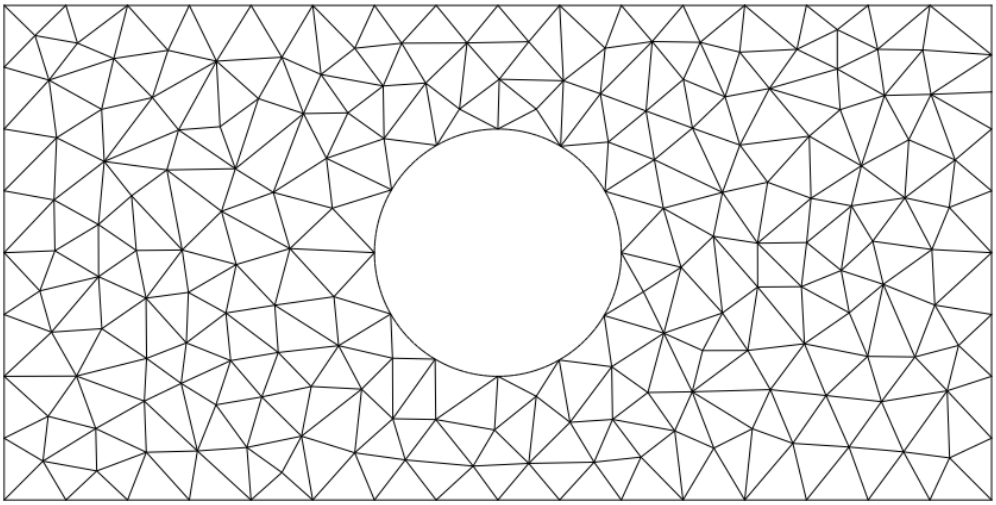
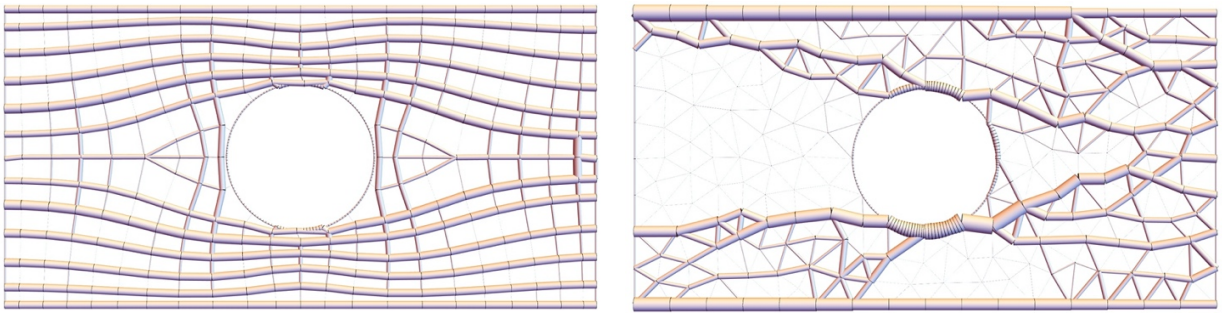


Figure 21. The lattice layout of the TLL model.

Before optimization, a radius of $0.1\ mm$ was assigned as the cross-sectional radius for all lattice members of both models. In this case, the SAL model had a volume of $31.188\ mm^3$ and a compliance of $3.7590\ \frac{mm}{N}$, while the TLL model had the same total volume and a compliance of $6.6108\ \frac{mm}{N}$, which is 75.9% higher than the SAL model's value. This illustrates the role that lattice member alignment to the load paths can have on the efficiency of a structure.

Figure 22 illustrates the results after cross-sectional optimization for the SAL model and the TLL model. We chose $130\ mm^3$ to be the target volume for the optimization, which is approximately 50% of the volume of the design domain with a $2\ mm$ thickness. The volume constraint is active

for both optimized designs from the SAL model and TLL model. The compliance of the SAL model after optimization is $0.5386\ \frac{mm}{N}$. The compliance of the TLL model after optimization is $0.6232\ \frac{mm}{N}$, which is 15.7% higher than the SAL model's value.



(a) (b)

Figure 22. (a) The stress-aligned lattice structure after running the cross-sectional area optimization, which has $0.5386\ \frac{mm}{N}$ *as the compliance value. (b) The arbitrary triangle lattice structure after running the cross-sectional area optimization, which has* $0.6232\ \frac{mm}{N}$ *as the compliance value.*

As shown in Figure 22, the lattices on the top and the bottom boundaries for the TLL model were generally larger than those in the SAL model. This means the interior lattice members for the SAL model are playing a larger role in providing support of the structure.

5. RESULTS AND DISCUSSION

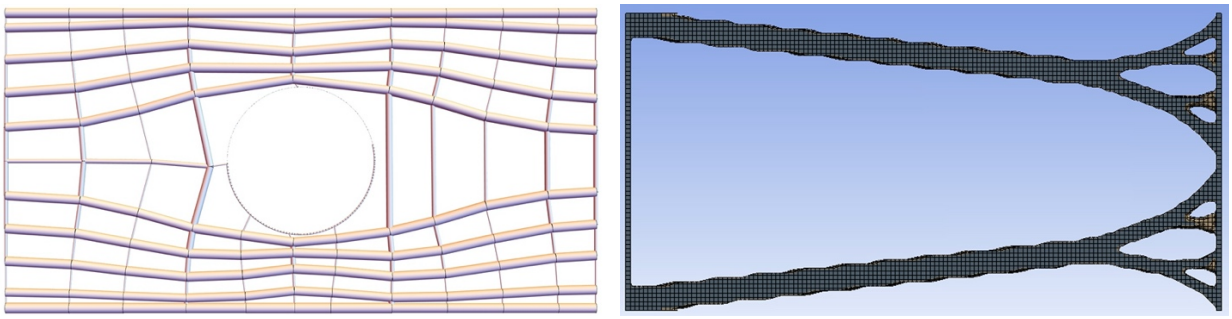
We illustrate the performance of the stress-aligned lattice (SAL) design approach on four different problems. In each case, the results are compared with those of topology optimization (TO). The three-dimensional TO results were obtained using ANSYS Workbench version 2019 [46]. The problems considered are: a plate with a hole under axial loading (PWH-Axial), a plate with a hole loaded as a cantilever beam (PWH-Cantilever), a plate with three-holes under axial loading, and a bridge problem. Both the SAL and TO design approaches solved these problems without assigning symmetric boundary condition to demonstrate their true performance. Each problem was solved for three different target volumes. In other words, in the second- (lower-) level optimization process for each problem, the lattice cross-sectional radii were optimized according to three different percentages of the maximum volume as the target total volume: 20%, 30%, and 40%. Each result is expressed in terms of a stiffness ratio, which is computed using Equation 8:

$$Stiffness\ Ratio = \frac{C_{solid}}{C_{lattice}} = \frac{1/S_{solid}}{1/S_{lattice}} = \frac{S_{lattice}}{S_{solid}} \qquad \textit{Equation 8}$$

where C_{solid} denotes the compliance of the original solid model that was computed using FEA, $C_{lattice}$ denotes the compliance of the optimized lattice model, under the specified loads for that problem, $S_{lattice}$ denotes the stiffness of the optimized lattice model, and S_{solid} denotes the stiffness of the original solid model. Hence, the larger stiffness ratio value represents a stiffer of the lattice model. We now show results of SAL on a number of case studies.

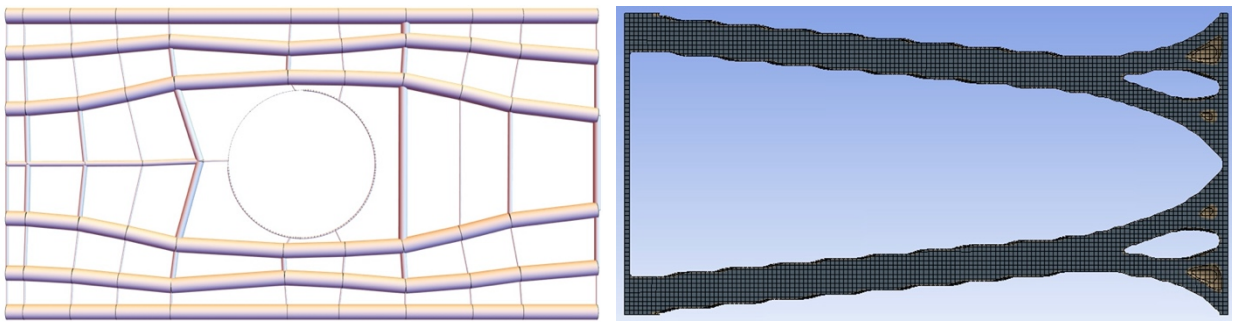
5.1. PLATE WITH A HOLE UNDER AXIAL LOADING

This is the same problem as described in Section 3. Figure 23 to Figure 25 shown the results of both the SAL method and the TO method with the different target volumes. These results show that the SAL algorithm is placing more lattice members and material along the primary load paths, and these results compare favorably with the TO designs qualitatively. Recall that the SAL algorithm behaves in a stochastic manner (see Sections 3.4 and 3.5), so a perfectly symmetrical lattice layout is not guaranteed.



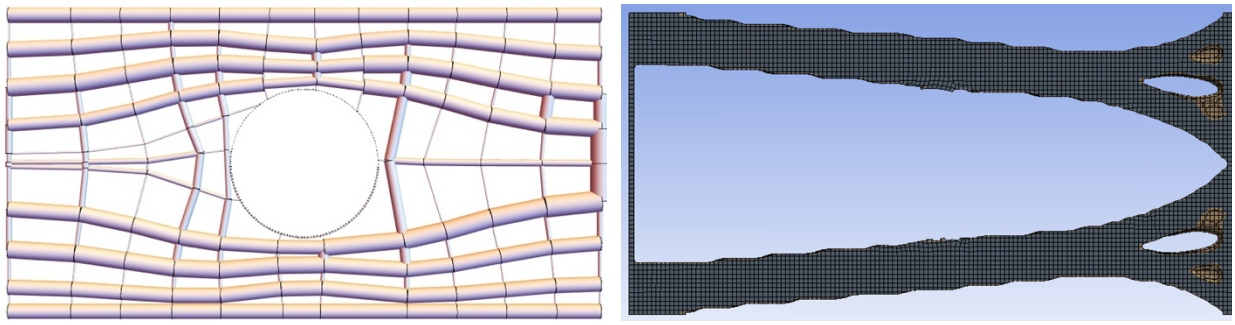
(a) (b)

Figure 23. The final result of the PWH-Axial problem with 20% target volume, generated with (a) the proposed SAL method and (b) the TO method using ANSYS Workbench version 2019. The stiffness ratio for (a) is 0.213 *and for (b) is* 0.178.



(a) (b)

Figure 24. The final result of the PWH-Axial problem with 30% target volume, generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.329 *and for (b) is* 0.284.



(a) (b)

Figure 25. The final result of the PWH-Axial problem with 40% target volume, generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.441 *and for (b) is* 0.408.

Figure 26 compares the stiffness ratios for the PWH-Axial optimized designs using both SAL and TO. The SAL and TO results compare favorably, though the difference between the two is slightly larger than expected. After significant investigation, the differences between these two design approaches can be attributed to several factors. The TO results are somewhat sensitive to the resolution of the mesh used in the underlying model. The model used to generate the results

in Figure 23 contains 32592 elements, with a mesh size of $0.3\ mm$. A finer mesh may result in a lower stiffness ratio, since decreasing the mesh size from $0.5\ mm$ to $0.3\ mm$ reduced the compliance by approximately 25%. Note that there are other differences between the two modeling approaches. The SAL model uses one-dimensional beam elements, while the TO results were obtained with three-dimensional solid elements. The beam element model has known errors at joints due to overlaps and gaps between the connected beam members. Finally, the cross-sectional shape of members in the SAL approach is circular, while members arising from the TO method can have any arbitrary cross-sectional shape.

Given all of the differences listed above, the reason for comparing the SAL results to those of the TO approach is not to demonstrate one or the other as superior, but rather to provide a comparison and validation of the current approach with known techniques. The differences between these two approaches over a wide range of problems deserves future study.

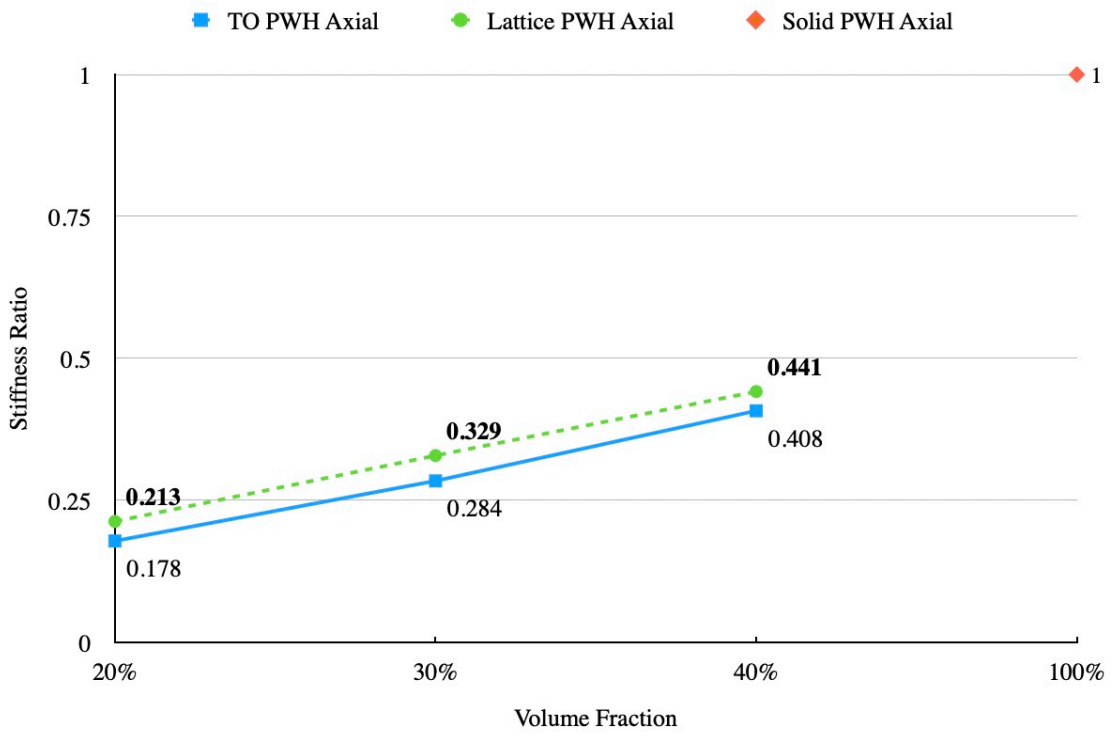
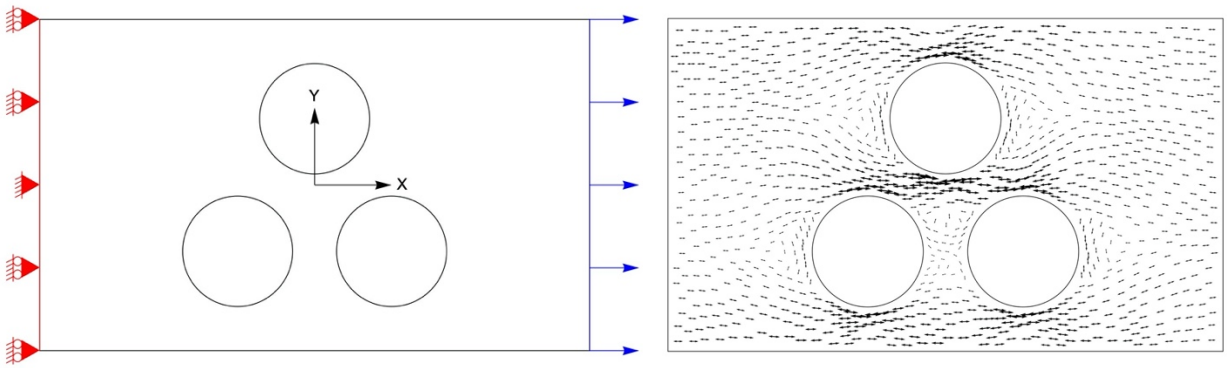


Figure 26. Stiffness ratio values versus Volume Fraction for the PWH-Axial problem. The green dashed line with circle markers represents the stiffness ratio values for the lattice models while the blue solid line with square markers represents the values for the TO models.

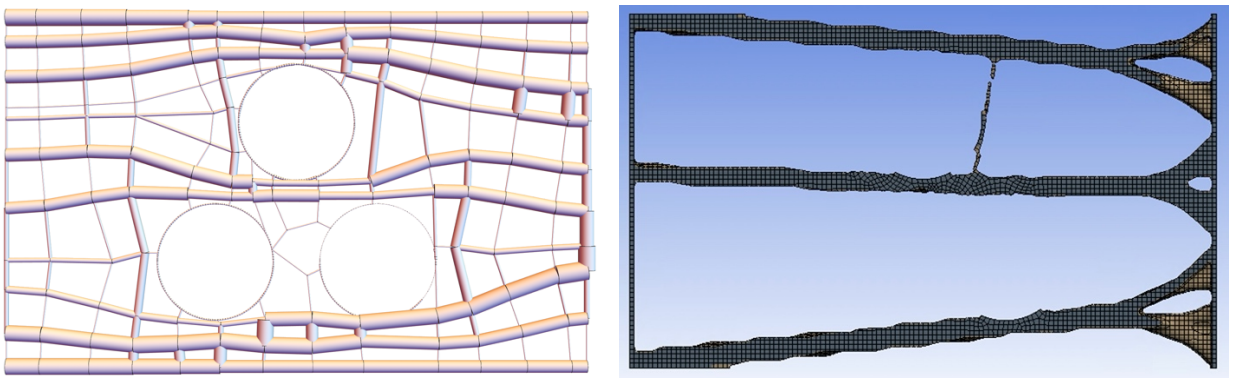
5.2. PLATE WITH THREE HOLES UNDER AXIAL LOADING

To test the approach on a problem with greater complexity of geometry and stress trajectories, we consider a plate of dimensions $50 \times 30 \times 2\ mm^3$ with three $5\ mm$ radius holes. The holes are located at $(0{,}6\ mm)$, $(-7\ mm, -6\ mm)$, and $(7\ mm, -6\ mm)$. The geometry and loading are illustrated in Figure 27(a), and the first principal stress field is shown in Figure 27(b). The design optimization results for the SAL method and the TO method are shown in Figure 28 to Figure 30.



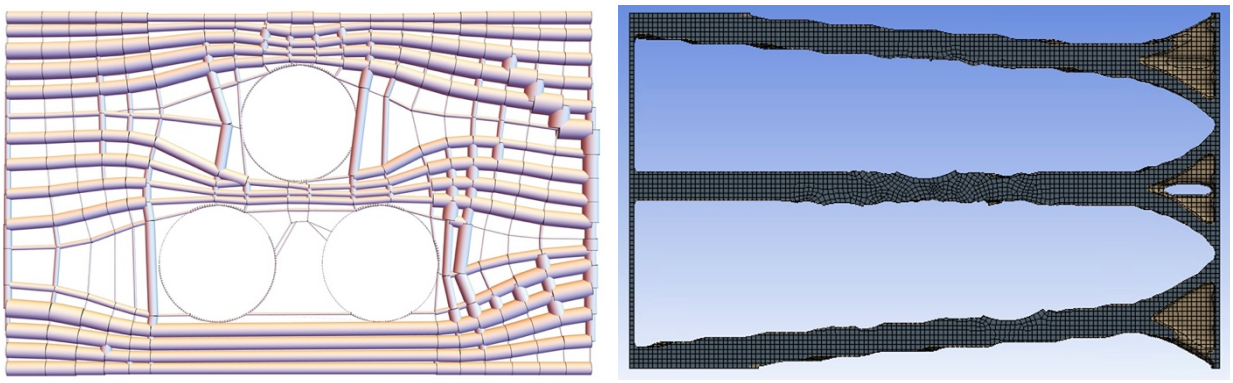
(a) (b)

Figure 27. (a) Geometry and loading for a plate with three holes under axial load, and (b) the 1st principal stress field.



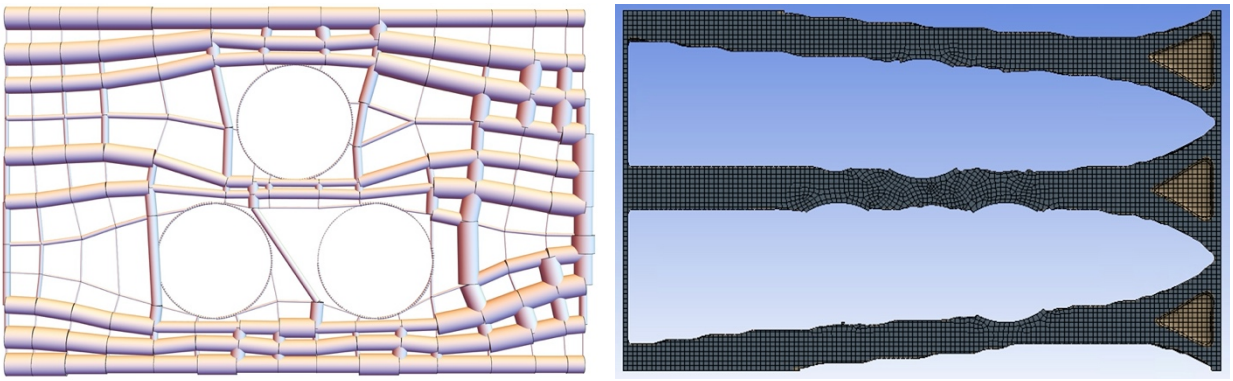
(a) (b)

Figure 28. Optimized designs for the three-hole problem with 20% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.205 *and for (b) is* 0.206.



(a) (b)

Figure 29. Optimized designs for the three-hole problem with 30% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.249 *and for (b) is* 0.351.



(a) (b)

Figure 30. Optimized designs for the three-hole problem with 40% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.333 *and for (b) is* 0.492.

Figure 31 compares the stiffness ratios for the SAL method and the TO method. As discussed above, the differences in these results can be attributed to the many differences between the two models, but the two sets of results are qualitatively very similar in terms of capturing the primary load paths. In this problem, the greater flexibility of the TO geometry to wrap around the circles

and to vary the cross-sectional beam shapes may provide a significant advantage compared to the circular cylinder lattice members in the SAL approach. A number of much better (lower compliance) designs were obtained with SAL than those shown in Figure 28 to Figure 30, but these are not presented here because the beam elements slightly penetrated the edge of one or more of the holes. This interference check was not part of the automated optimization process; it is anticipated that adding these checks would allow the optimizer to find better SAL solutions than those shown here, though likely still not as good as those found with TO.

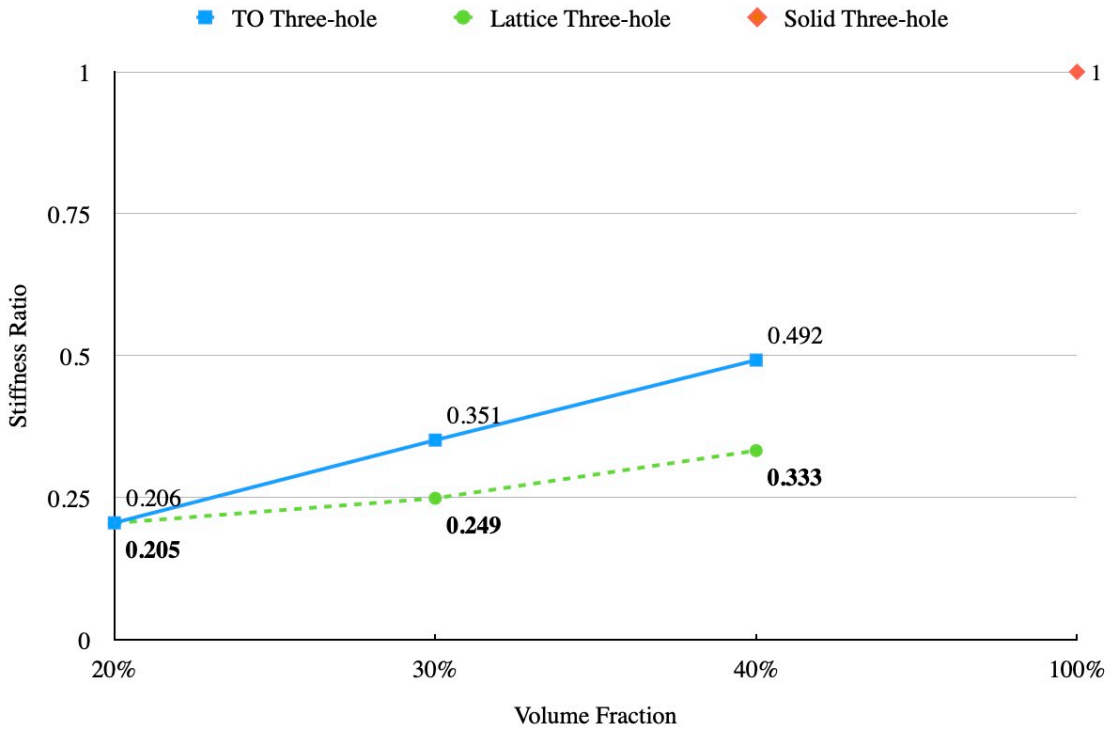
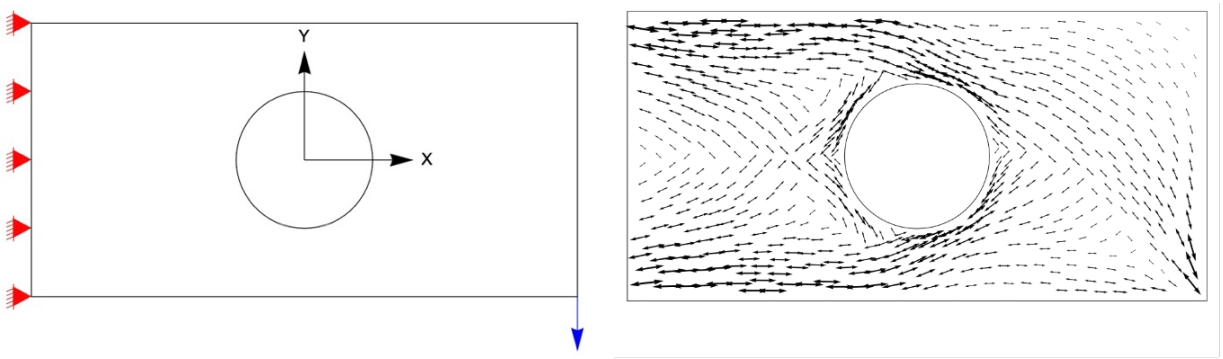


Figure 31. Stiffness ratio values versus Volume Fraction for the three-hole plate problem. The green dashed line with circle markers represents the stiffness ratio values for the lattice models while the blue solid line with square markers represents the values for the TO models.

5.3. PLATE WITH A HOLE UNDER BENDING

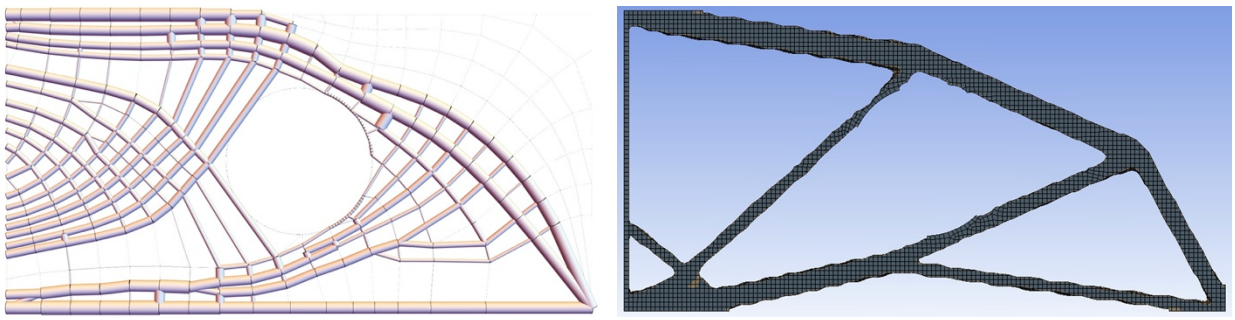
This problem has an identical geometry as the original example problem, with a $40 \times 20 \times 2\ mm^3$ design domain, and a hole in the center of the domain with radius $5\ mm$. All DOFs of the left boundary are fixed, and a vertical load is applied to the lower right corner of the domain. The problem geometry and the first principal stress field are shown in Figure 32. Results from both the SAL method and the TO method are shown in Figure 33 to Figure 35.

The material layout and stiffness ratios for the SAL and the TO designs are very consistent for the volume fractions of 30% and 40%, but the designs are somewhat different for the volume fraction of 20%.



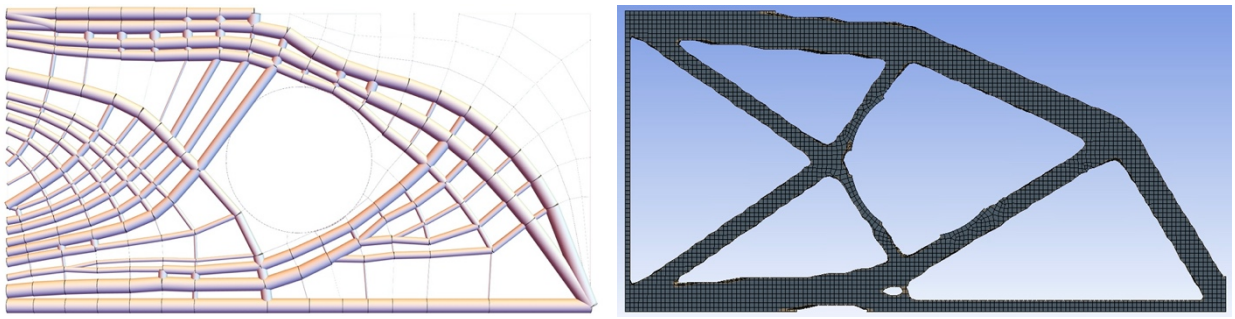
(a) (b)

Figure 32. (a) The problem setup for the PWH-cantilever problem and (b) the 1st principal stress direction field.



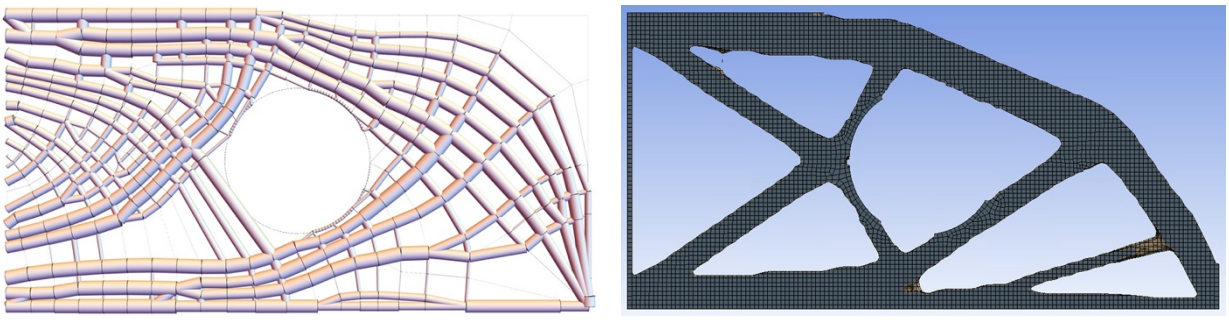
(a) (b)

Figure 33. Optimized designs for the PWH-cantilever problem with a 20% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.290 *and for (b) is* 0.238.



(a) (b)

Figure 34. Optimized designs for the PWH-cantilever problem with a 30% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.430 *and for (b) is* 0.416.



(a) (b)

Figure 35. Optimized designs for the PWH-cantilever problem with a 40% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.537 *and for (b) is* 0.560.

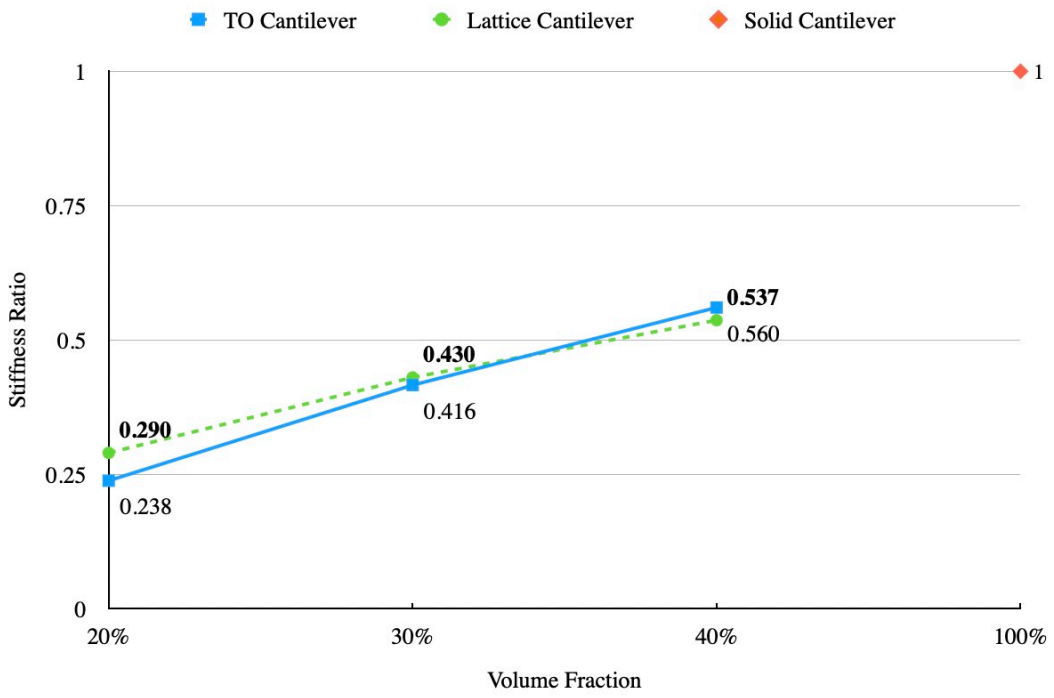
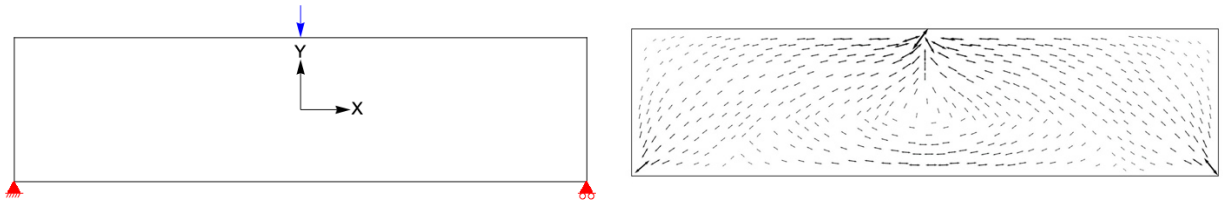


Figure 36. Stiffness ratio values versus Volume Fraction for the PWH-Cantilever problem. The green dashed line with circle markers represents the stiffness ratio values for the lattice models while the blue solid line with square markers represents the values for the TO models.

5.4. BRIDGE PROBLEM

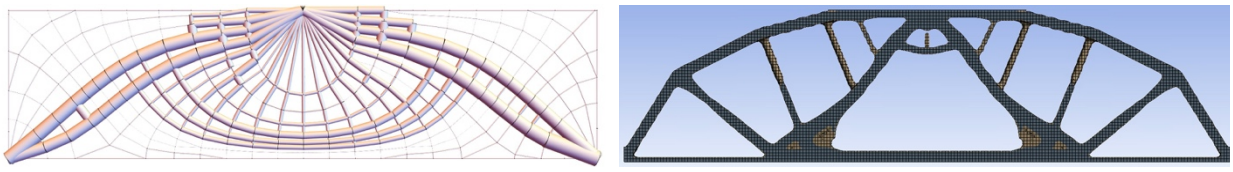
A classic bridge optimization problem is selected as the final example problem. The design domain for this problem is a $80 \times 20 \times 2\ mm^3$ cuboid. The boundary conditions and loading for this problem are shown in Figure 37(a). All DOFs at the lower left corner are fixed while only y-DOF is fixed at the lower right corner, a vertical load is applied at the middle of the top boundary. The 1^{st} principal stress field for this problem is shown in Figure 37(b).



(a) (b)

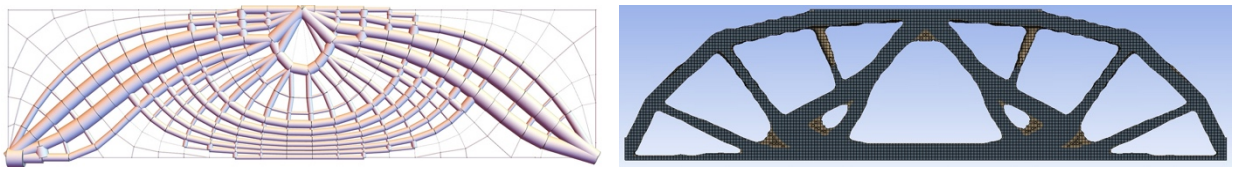
Figure 37. (a) Problem definition for the bridge problem and (b) the 1^{st} principal stress field.

Figure 38 to Figure 40 contain the optimized designs of the SAL and TO methods for volume fractions of 20%, 30%, and 40%. The SAL and TO results compare favorably from a qualitative perspective.



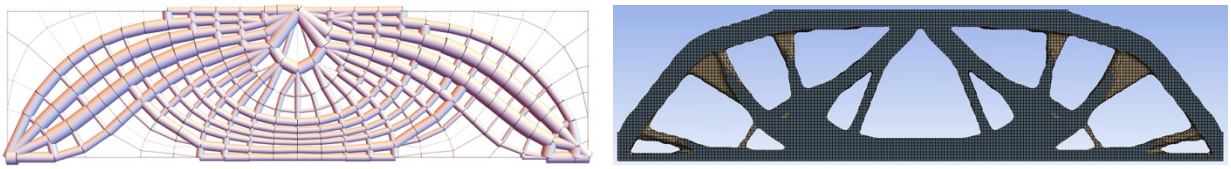
(a) (b)

Figure 38. Optimized designs for the bridge problem with 20% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.589 *and for (b) is* 0.226.



(a) (b)

Figure 39. Optimized designs for the bridge problem with 30% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.763 *and for (b) is* 0.374.



(a) (b)

Figure 40. Optimized designs for the bridge problem with 40% target volume generated with (a) the SAL method and (b) the TO method. The stiffness ratio for (a) is 0.947 *and for (b) is* 0.510.

Figure 41 compares the stiffness ratios for the SAL method and the TO method. In this problem, the SAL designs have lower compliance than the TO designs. In Figure 38 to Figure 40 it is clear that the SAL designs have thick lattice members near the point where the load is applied. As discussed previously, beam models do not account for material gaps or overlaps at joints. Thus, the overlapping lattices at regions vicinity of the point load and point fix boundary provide extra amount of stiffness beyond physical limitation due to geometric of representation of beam models. The SAL designs require some post-processing (not performed here) to provide an accurate representation of the joint geometry and structural performance. However, the similarity of material arrangement of the SAL designs compared to the TO designs still proves the great performance of the SAL method.

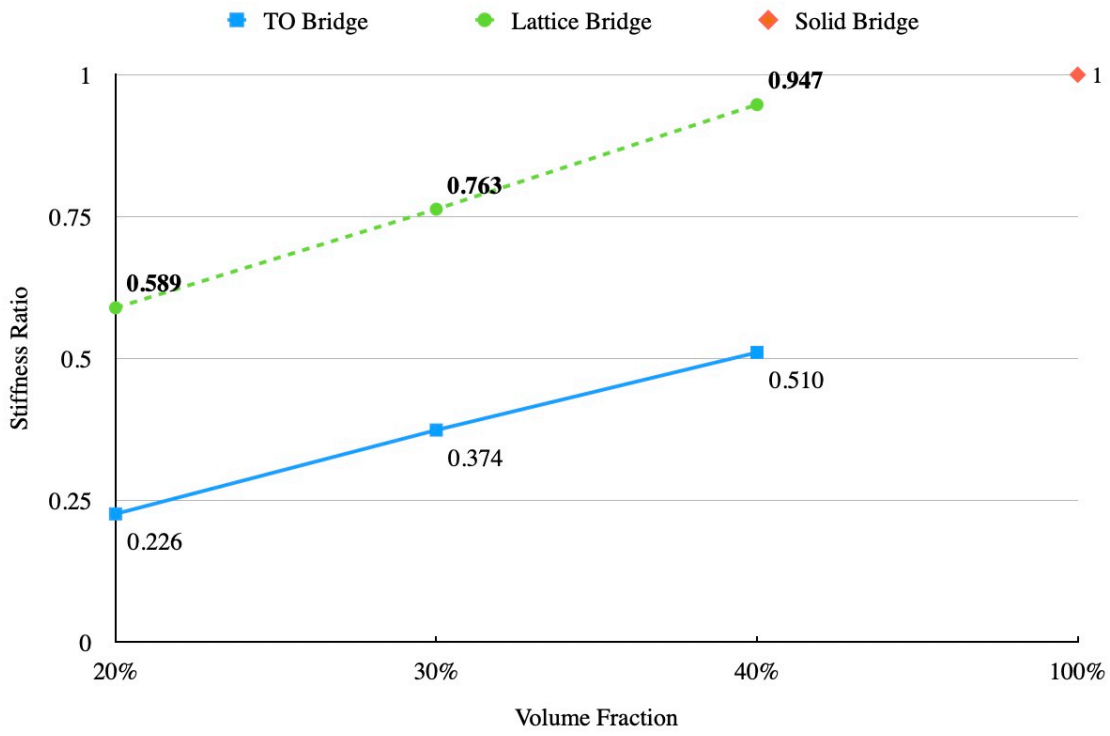


Figure 41. Stiffness ratio values versus Volume Fraction for the bridge problem. The green dashed line with circle markers represents the stiffness ratio values for the lattice models while the blue solid line with square markers represents the values for the TO models.

6. CONCLUSIONS

The novel lattice design optimization method presented here has produced light-weight structural designs by aligning the lattice members with the primary load paths defined by the principal stress fields. By simultaneously performing layout and sizing optimization using a bi-level design optimization approach on a number of proof-of-principle structural design problems, the design space has been explored more effectively and higher performing designs have been obtained by the proposed method. This approach can be applied to multi-physics problems involving thermal conduction [47] and other field problems with only minor modification.

7. FUTURE WORK

Future studies are needed in several areas, the critical one is to extend the adaptability for the stress-aligned lattice design method from a 2D to 3D environment. The current method has a limitation in connecting the stress trajectories efficiently when the 3^{rd} principal stress is involved in a 3D problem. This issue may be able to be resolved by only generating the 1^{st} principal stress trajectories and connect the nearby trajectories by short straight lines to form a local triangular or tetrahedron layout. Such approaches ensure the lattice to be generated along the most critical stress direction while the local triangular or tetrahedron layout enhances the structural stiffness.

The compliance computation accuracy for the final lattice model should be increased. As stated in section 5, the final lattice model has each lattice member connected to others directly as a node-to-node structure, also, some lattice members are overlapping around the region that has a high local stress value. Such issues can be resolved by adding a solid sphere joint on every node to smoothen the nodal connection, then output the model as a single solid 3D model and apply FEA to such a solid model to compute the compliance value. A compliance value computed in this way should better describe the performance of the stress-aligned lattice design.

The initial (baseline) values for the lattice cross-sectional radii could be generated at the end stage of the lattice layout generation process based on the lattice members' local stress state. This approach might increase the efficiency for the 2nd (lower) level optimization which optimizes the lattice radii compared to the current method, in which one small value, that is $0.1\ mm$, is assumed as the initial value for all lattice radii.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] F. Libonati and M. J. Buehler, "Advanced structural materials by bioinspiration," *Advanced Engineering Materials*, vol. 19, no. 5, p. 1600787, 2017.

[2] I. W. Hamley, "Protein assemblies: nature-inspired and designed nanostructures," *Biomacromolecules*, vol. 20, no. 5, pp. 1829–1848, 2019.

[3] A. B. Wardrop, "Evidence for the possible presence of a microtrabecular lattice in plant cells," *Protoplasma*, vol. 115, no. 1, pp. 81–87, 1983.

[4] H. N. G. Wadley and D. T. Queheillalt, "Thermal applications of cellular lattice structures," in *Materials science forum*, 2007, vol. 539, pp. 242–247.

[5] K. Saha, S. Acharya, and C. Nakamata, "Heat transfer enhancement and thermal performance of lattice structures for internal cooling of airfoil trailing edges," *Journal of Thermal Science and Engineering Applications*, vol. 5, no. 1, 2013.

[6] S. Catchpole-Smith, R. R. J. Sélo, A. W. Davis, I. A. Ashcroft, C. J. Tuck, and A. Clare, "Thermal conductivity of TPMS lattice structures manufactured via laser powder bed fusion," *Additive Manufacturing*, vol. 30, p. 100846, 2019.

[7] X. Bai, Z. Zheng, and A. Nakayama, "Heat transfer performance analysis on lattice core sandwich panel structures," *International journal of heat and mass transfer*, vol. 143, p. 118525, 2019.

[8] F. Scarpa, M. Ouisse, M. Collet, and K. Saito, "Kirigami auxetic pyramidal core: mechanical properties and wave propagation analysis in damped lattice," *Journal of Vibration and Acoustics*, vol. 135, no. 4, 2013.

[9] D. T. Queheillalt and H. N. G. Wadley, "Cellular metal lattices with hollow trusses," *Acta Materialia*, vol. 53, no. 2, pp. 303–313, 2005.

[10] Z.-J. Wu, F.-M. Li, and C. Zhang, "Vibration band-gap properties of three-dimensional Kagome lattices using the spectral element method," *Journal of Sound and Vibration*, vol. 341, pp. 162–173, 2015.

[11] E. Baravelli and M. Ruzzene, "Internally resonating lattices for bandgap generation and low-frequency vibration control," *Journal of Sound and Vibration*, vol. 332, no. 25, pp. 6562–6579, 2013.

[12] W. Beugeling, N. Goldman, and C. M. Smith, "Topological phases in a two-dimensional lattice: Magnetic field versus spin-orbit coupling," *Physical Review B*, vol. 86, no. 7, p. 75118, 2012.

[13] V. S. Deshpande, N. A. Fleck, and M. F. Ashby, “Effective properties of the octet-truss lattice material,” *Journal of the Mechanics and Physics of Solids*, vol. 49, no. 8, pp. 1747–1769, 2001.

[14] T. Tancogne-Dejean and D. Mohr, “Elastically-isotropic truss lattice materials of reduced plastic anisotropy,” *International Journal of Solids and Structures*, vol. 138, pp. 24–39, 2018.

[15] G. N. Labeas and M. M. Sunaric, “Investigation on the static response and failure process of metallic open lattice cellular structures,” *Strain*, vol. 46, no. 2, pp. 195–204, 2010.

[16] G. M. Tassi, Q. Gao, and A. R. Diaz, “Infill Analysis and Optimization in Additive Manufacturing Applications,” in *International Conference on Engineering Optimization*, 2018, pp. 593–604.

[17] F. Campagna and A. R. Diaz, “Optimization of lattice infill distribution in additive manufacturing,” 2017.

[18] M. C. Messner, “Optimal lattice-structured materials,” *Journal of the Mechanics and Physics of Solids*, vol. 96, pp. 162–183, 2016.

[19] A. Challapalli and J. Ju, “Continuum model for effective properties of orthotropic octet-truss lattice materials,” in *ASME International Mechanical Engineering Congress and Exposition*, 2014, vol. 46583, p. V009T12A051.

[20] E. Ptochos and G. Labeas, “Elastic modulus and Poisson’s ratio determination of micro-lattice cellular structures by analytical, numerical and homogenisation methods,” *Journal of Sandwich Structures & Materials*, vol. 14, no. 5, pp. 597–626, 2012.

[21] J. C. Wallach and L. J. Gibson, “Mechanical behavior of a three-dimensional truss material,” *International Journal of Solids and Structures*, vol. 38, no. 40–41, pp. 7181–7196, 2001.

[22] I. Maskery *et al.*, “An investigation into reinforced and functionally graded lattice structures,” *Journal of Cellular Plastics*, vol. 53, no. 2, pp. 151–165, 2017.

[23] Y. Wang, L. Zhang, S. Daynes, H. Zhang, S. Feih, and M. Y. Wang, “Design of graded lattice structure with optimized mesostructures for additive manufacturing,” *Materials & Design*, vol. 142, pp. 114–123, 2018.

[24] S. Daynes, S. Feih, W. F. Lu, and J. Wei, “Optimisation of functionally graded lattice structures using isostatic lines,” *Materials & Design*, vol. 127, pp. 215–223, 2017.

[25] D. W. Kelly and M. W. Tosh, “Interpreting load paths and stress trajectories in elasticity,” *Engineering Computations*, vol. 17, no. 2, pp. 117–135, 2000.

[26] R. Weijermars, “Mapping stress trajectories and width of the stress-perturbation zone near a cylindrical wellbore,” *International Journal of Rock Mechanics and Mining Sciences*, vol. 64, pp. 148–159, 2013.

[27] S. Daynes, S. Feih, W. F. Lu, and J. Wei, “Design concepts for generating optimised lattice structures aligned with strain trajectories,” *Computer Methods in Applied Mechanics and Engineering*, vol. 354, pp. 689–705, 2019.

[28] J. Wu, W. Wang, and X. Gao, “Design and Optimization of Conforming Lattice Structures,” *arXiv preprint arXiv:1905.02902*, 2019.

[29] A. Sinha, P. Malo, and K. Deb, “A review on bilevel optimization: from classical to evolutionary approaches and applications,” *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 2, pp. 276–295, 2017.

[30] P. Rahkola, “Parameter optimisation using Heeds MDO and DAKOTA,” 2014.

[31] S. Wolfram and others, *The MATHEMATICA® book, version 4*. Cambridge university press, 1999.

[32] J. E. Marsden and T. J. R. Hughes, *Mathematical foundations of elasticity*. Courier Corporation, 1994.

[33] D. J. Higham and N. J. Higham, *MATLAB guide*. SIAM, 2016.

[34] K. Svanberg, “The method of moving asymptotes—a new method for structural optimization,” *International journal for numerical methods in engineering*, vol. 24, no. 2, pp. 359–373, 1987.

[35] A. O. Bochkarev and M. A. Grekov, “Influence of surface stresses on the nanoplate stiffness and stability in the Kirsch problem,” *Physical Mesomechanics*, vol. 22, no. 3, pp. 209–223, 2019.

[36] C. Kirsch, “Die theorie der elastizitat und die bedurfnisse der festigkeitslehre,” *Zeitschrift des Vereines Deutscher Ingenieure*, vol. 42, pp. 797–807, 1898.

[37] D. L. Logan, *A first course in the finite element method*. Cengage Learning, 2011.

[38] E. Russell Johnston, *Mechanics of materials*. Copyright, 2012.

[39] D. Li, W. Liao, N. Dai, and Y. M. Xie, “Anisotropic design and optimization of conformal gradient lattice structures,” *Computer-Aided Design*, vol. 119, p. 102787, 2020.

[40] F. L. Teixeira and W. C. Chew, “Lattice electromagnetic theory from a topological viewpoint,” *Journal of mathematical physics*, vol. 40, no. 1, pp. 169–187, 1999.

[41] K. E. Atkinson, *An introduction to numerical analysis*. John wiley & sons, 2008.

[42] O. J. B. A. Pereira and J. P. B. M. de Almeida, “Automatic drawing of stress trajectories in plane systems,” *Computers & structures*, vol. 53, no. 2, pp. 473–476, 1994.

[43] W. H. Greub, *Linear algebra*, vol. 23. Springer Science & Business Media, 2012.

[44] M. Imada, S. Noda, A. Chutinan, T. Tokuda, M. Murata, and G. Sasaki, “Coherent two-dimensional lasing action in surface-emitting laser with triangular-lattice photonic crystal structure,” *Applied physics letters*, vol. 75, no. 3, pp. 316–318, 1999.

[45] M. Qiu and B. Jaskorzynska, “Design of a channel drop filter in a two-dimensional triangular photonic crystal,” *Applied Physics Letters*, vol. 83, no. 6, pp. 1074–1076, 2003.

[46] H.-H. Lee, *Finite element simulations with ANSYS Workbench 18*. SDC publications, 2018.

[47] Z. Jiang, Q. Gao, R. C. Averill, K. Deb, and E. D. Goodman, “Field-Aligned Lattice Design for Thermal Problems.”