A Globally Conforming Lattice Structure for 2D Stress Tensor Visualization

Wang, Junpeng; Wu, Jun; Westermann, Rüdiger

DOI
10.1111/cgf.13991

Publication date
2020

Document Version
Accepted author manuscript

Published in
Computer Graphics Forum

Citation (APA)
Wang, J., Wu, J., & Westermann, R. (2020). A Globally Conforming Lattice Structure for 2D Stress Tensor Visualization. *Computer Graphics Forum*, *39*(3), 417-427. https://doi.org/10.1111/cgf.13991

Volume xx (200y), Number z, pp. 1–11

A Globally Conforming Lattice Structure for 2D Stress Tensor Visualization

Junpeng Wang[1], Jun Wu[2] & Rüdiger Westermann[1]

[1] Chair for Computer Graphics and Visualization, Technical University of Munich, Germany
[2] Department of Design Engineering, Delft University of Technology, The Netherlands

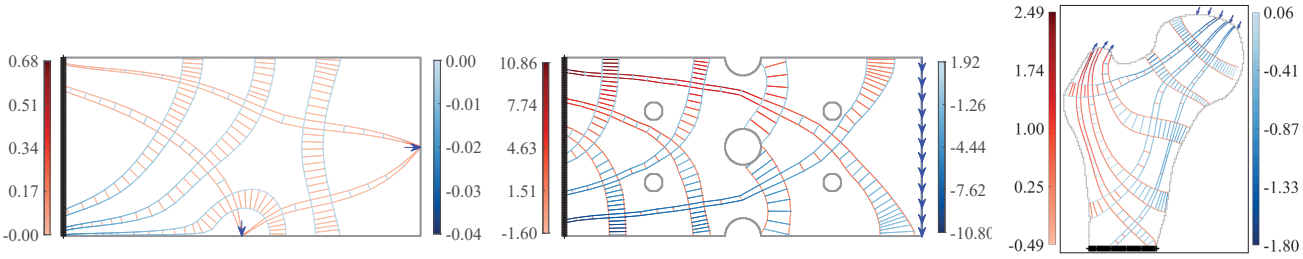

Figure 1: *Visualization of stress tensor fields in 2D solid objects under load using the conforming lattice. From left to right: Plate, perforated plate, slice through a femur. Arrows indicate the applied loads and bold black lines the fixation regions. Stress direction, convergence and divergence is shown by principal stress lines along the major (red) and minor (blue) principal directions. Stress ratio is shown by the elements' shape. Tension and compression is encoded into shades of red and blue.*

Abstract

We present a visualization technique for 2D stress tensor fields based on the construction of a globally conforming lattice. Conformity ensures that the lattice edges follow the principal stress directions and the aspect ratio of lattice elements represents the stress anisotropy. Since such a lattice structure cannot be space-filling in general, it is constructed from multiple intersecting lattice beams. Conformity at beam intersections is ensured via a constrained optimization problem, by computing the aspect ratio of elements at intersections so that their edges meet when continued along the principal stress lines. In combination with a coloring scheme that encodes relative stress magnitudes, a global visualization is achieved. By introducing additional constraints on the positional variation of the beam intersections, coherent visualizations are achieved when external loads or material parameters are changed. In a number of experiments using non-trivial scenarios, we demonstrate the capability of the proposed visualization technique to show the global and local structure of a given stress field.

1. Introduction

Techniques for visualizing the stress distribution in solid objects under load are important in a number of applications ranging from lightweight structure design over implant planning to the design of support structures. Such visualizations improve our understanding of the material response to external load conditions, and give rise to improved object designs regarding their mechanical properties.

At each point in a stress field, the state of stress is fully described by the stress vectors for three mutually orthogonal orientations of a differential area element at that point. From these orientations, the so-called principal stresses can be computed, i.e., the normal stresses into the directions where the shear stress components vanish. These normal stresses, which include the maximum and minimum normal stress components acting at a point, are fundamental to the visualization of stress tensor fields. Their visualization, however, is challenging due to several reasons:

- It requires to find visual abstractions of the stress tensor, to convey the principal stress directions and magnitudes in a meaningful way. This includes distinguishing between the different types of normal stresses, i.e., tension and compression, as well as the ratio of the principal stresses. This information should be visualized simultaneously, to reveal the mutual variations of the principal stresses across the solid body.
- The visualization should provide a global view of the stress field,

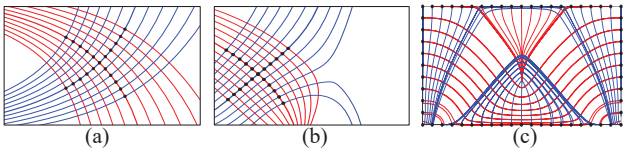

Figure 2: *(a, b) Minor Principal Stress Lines (PSLs) (blue) are seeded at equally spaced seed points (•) along the initial PSL (bold orange). Additional major PSLs (orange) are then seeded along one of the minor PSLs (bold blue). (a) The resulting grid cells do not represent the local stress ratio. (b) The domain is incompletely covered. (c) PSLs concentrate despite uniform seeding density along the domain boundaries.*

to convey a general impression of the major mechanical properties of the body under load, as well as their spatial dependencies under varying load conditions and when shape or topology variations are applied.
- It is desired to show continuous stress trajectories, i.e., the principal stress lines (PSLs), that reveal along which paths external loads are transmitted. This supports finding paths along which loads are predominantly transmitted from one boundary to another, to analyse how load transmission is affected by variations in the structure of the simulated material and external load conditions, and to indicate where the reduction of distances between the PSLs come along with a simultaneous increase of stresses.

In engineering, the most common visualization of 2D tensor fields is by means of so-called trajectory images, which show selected PSLs in the domain [Tim83, Fro48]. Such visualizations are generated by selecting an initial seed point to start the PSLs and placing new seed points automatically along the initial trajectory, or by seeding uniformly along the object's boundary. Even though this kind of visualization can provide a global view on the stress distribution, it has weaknesses if PSLs are not selected carefully. As shown in Fig. 2, the resulting grid structure usually does not convey the local stress state, since the size and aspect ratio of the generated grid cells is dictated by the initial seeding strategy. Furthermore, such visualizations can result in strongly varying trajectory density, which can mislead the interpretation of stress concentration.

1.1. Contribution

We propose a visualization technique for 2D stress fields using PSLs, which considers the aforementioned requirements and overcomes some of the limitations of classical trajectory images. We introduce the *globally conforming lattice*, a grid structure that aligns with the principal stress directions. Yet it is not domain-filling but comprised of quadrilateral (2D) elements aligned along the principal stress directions, so called beams. Beams are selected interactively, and the geometry of the beam elements is constructed so that they convey the anisotropy of the principal stresses. In 2D, where beams intersect and share an element at the intersection, this element has to conform the geometry of both beams. By ensuring conformity at all beam intersections, a globally conforming structure is generated. Our method builds upon the following specific contributions:

- We introduce the use of beams instead of single lines to create a stress-following grid structure in multiple dimensions, and to encode the ratio of principal stresses into the geometry of the beam elements. Thus, only line segments coinciding with stress lines are shown, along all principal directions.
- Conformity of beams at intersections is achieved via the solution of a constrained optimization problem. The optimization computes for all intersection points the size and aspect ratio of corresponding beam elements, so that the ratio of principal stresses is maintained and the edges of connected elements meet when continued along the respective PSL.
- We provide different color mappings for points along the trajectories to distinguish between tension and compression or the relative stress magnitudes along the principal stress directions.

For different solids and load conditions, we demonstrate the capability of our method to provide a globally conforming visualization of a 2D stress distribution. We further demonstrate the use of this method for visualizing 3D tensor fields. In 3D, PSLs do not intersect, in general, so that forming a beam structure with cycles as in 2D becomes unfeasible. Thus, we let the user interactively select seed positions and progressively grow new beams composed of hexahedral elements along the PSLs. The growth process considers the design decisions underlying the construction of a conforming lattice in 2D, and, thus, adheres to the identified requirements.

2. Related work

Besides the use of trajectory images, 2D stress tensor fields can be visualized in a number of different ways, each coming with its own strength and weakness. Let us refer here to the work by Kratz et al. [KASH13], which provides a thorough discussion of the properties of many different stress visualization techniques.

For an overview of the stress state in a solid object, one often resorts to the visualization of scalar stress measures like the von Mises stress [DGBW09, KMH11] or the material's index of refraction when loaded [BES15]. Such measures are derived from the components of the stress tensor, and they can be visualized using standard techniques like direct volume rendering or iso-contouring. Yet since these techniques simplify the complex stress state to a single scalar number, they cannot accurately convey the shape of load transmission pathways. In particular, the mutual dependencies between major and minor stresses—which are important for a structural stress analysis of a solid under load—are lost.

Another alternative to visualize stress tensor fields is by means of tensor glyphs, i.e., geometric primitives encoding tensor invariants by visual attributes like shape and color. Tensor glyphs originate from early work on glyph-based diffusion tensor visualization [Kin04], and a number of variants have been designed for the visualization of positive definite tensors [KWS*08], general symmetric tensors [SK10], as well as non-symmetric tensors [ZP05, SK16, GRT17]. Placement strategies for dense glyph visualizations help to reduce visual clutter [KW06, HSH07], and special glyph designs have been proposed to support the comparative visualization of diffusion tensors [ZYLL08].

While tensor glyphs can effectively convey the local stress state, due to their discrete nature they make it difficult to accurately infer

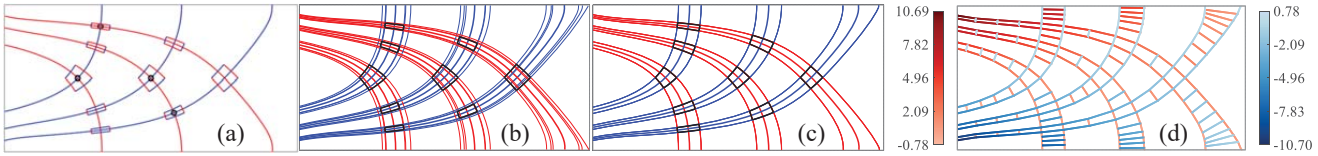

Figure 3: *Method overview: (a) Selected nodes (black circles) and computed intersections, skeleton trajectories, seed elements oriented along PSLs. (b) Edges of connected nodes do not lay on the same PSL, (c) but do so due to anisotropic scaling during optimization. (d) Conforming lattice, where beams of elements connect the seed elements. Coloring indicates stress magnitude, tension and compression.*

the global shape and the divergent or convergent behavior of load transmission pathways. In contrast, the technique we propose aims at encoding the local stress state by the cells of a conforming grid, and achieving a continuous impression—also revealing the global relationships between induced loads and material response—by growing these cells along PSLs.

To generate the conforming grid, we build upon the computation of continuous stress trajectories by using Lagrangian particle tracing in the principal stress direction fields. In the work by Delmarcelle and Hesselink [DH93], stress trajectories are used to generate so-called hyperstreamlines. A hyperstreamline shows a cylinder-like geometric structure, which is formed by extruding ellipses along a selected major PSL. The major- and minor-axis of the ellipses correspond to the direction and length of the eigenvectors and -values of the medium and minor stresses. Even though hyperstreamlines were introduced for the visualization of 3D stress fields, they can be adapted straightforwardly to 2D scenarios.

Silhouettes and ridges of a hyperstreamline, however, do not coincide with stress lines, possibly misleading the user in the interpretation of the underlying stress field. The spatial extent of hyperstreamlines prohibits placing them close to each other, making it difficult to reveal contracting behaviour of the PSLs. In 3D, it is also difficult to extract the orientation of a hyperstreamline when the medium and minor stresses have similar magnitude. By using a grid structure that is solely composed of (segments) of PSLs, we overcome these limitations.

Visualizations building solely upon PSLs have also been proposed in previous works. Stress-nets [WB05] are obtained by rendering together major and minor PSLs, at the same time trying to place them evenly to reduce clustering. Dick et al. [DGBW09] select random seed points in the 3D domain and trace PSLs along all principal directions, simultaneously encoding tension and compression by color. Both approaches, due to the random selection of PSLs, face the same problems as classical trajectory images and cannot adhere to our specified requirements in general. For surface remeshing, Alliez et al. [ACSD*03] build an initial control mesh that follows the principal curvature directions, derived from the curvature tensor. When applied to stress tensor fields, this approach produces visualizations similar to trajectory images, and cannot effectively control the shape and local density of elements. Hotz et al. [HFH*06] smear out dye along along the PSLs using line integral convolution, to generate a density field that resembles a grid-like structure, This approach provides a global overview of the stress distribution, yet it cannot accurately reveal the local stress state and does not produce continuous load transmission pathways.

Besides visualizing the directional information in a stress field, a number of works have studied the topology of symmetric 2D and 3D tensor fields [DH94, HLL97]. They characterize the topology of a tensor field by degenerate structures where two or more eigenvalues are equal. The robust extraction of the topological skeleton using numerical schemes has been addressed by Zheng and Pang [ZP04] and Roy et al. [RKZZ18]. Topological approaches are different to our approach, since they focus on the extraction of specific points or surfaces where the eigenvector fields behave in a specific way. Let us refer to the works by Zobel and Scheuermann [ZS18] and Raith et al. [RBN*19] for thorough overviews of this field.

3. Mechanical foundations and method overview

In the following, we first describe the mechanical foundations underlying the computation of stress trajectories, and then briefly summarize how our proposed technique makes use of these trajectories. Since the fundamental relationships required for the computation of stress trajectories in 2D solids can be derived as special cases of the 3D case, we focus on the latter case in the following.

3.1. Trajectories in mechanics

In a material under load, the point-wise stress vector σ is defined as $\sigma = dF/dA$, where dF is the force which the material on one side of an infinitely small area element dA exerts on the material on the other side. At each point, the state of stress is fully described by the stress vectors for three mutually orthogonal orientations of the area element. In particular, the second-order stress tensor S, represented by a 3×3 matrix, contains the stress vectors for the three orientations corresponding to the axes of a Cartesian coordinate system:

$$S = \begin{bmatrix} \sigma_x & \tau_{yx} & \tau_{zx} \\ \tau_{yx} & \sigma_y & \tau_{zy} \\ \tau_{zx} & \tau_{zy} & \sigma_z \end{bmatrix} \quad (1)$$

Here, the mixed index components correspond to the shear stresses, and they are equal on mutually orthogonal planes.

For an arbitrary orientation of the area element specified by its normal vector n, the stress vector is determined by Sn. This vector can be decomposed into a normal stress and a shear stress component, acting orthogonally and tangentially on the area element, respectively. For each stress tensor, there are three mutually orthogonal orientations of the area element where the shear stress components vanish. For these orientations, the normal stresses are called the principal stresses of the stress tensor.

The solution of the eigenvalue problem for S results in the three eigenvalues, which represent—sorted in descending order—the principal stresses σ_1, σ_2, and σ_3. If the eigenvalues are different, the corresponding eigenvectors are linearly independent and even mutually orthogonal due to the symmetry of S. The eigenvectors have unique direction, yet their orientation cannot be decided in general.

For an arbitrary start point, the PSLs are computed by a numerical integration scheme over the selected eigenvector field. We use a Runge-Kutta RK2(3) scheme with fixed integration step size δ. Since there is no consistent orientation of the eigenvectors, rather than interpolating eigenvectors, we interpolate the stress state in 2D in the form σ_x, σ_y and τ_{xy}. From the interpolated quantities, the principal stress components are derived.

3.2. Method overview

Our method starts with a discrete grid structure on which stress tensors are given. It needs to be equipped with an interpolation scheme, so that continuous stress trajectories can be computed. In addition, we provide the user information about the location of degenerate points where multiple eigenvalues are equal and the PSLs can cross. This guides the user towards regions where the generated grid structure has high deformation.

Selection: The user interactively picks points in the domain. If a selected location is close to an existing trajectory, the point is snapped to that trajectory. For each point, the PSL passing through that point (if not available from a previous selection) and the intersection points between the new and existing trajectories are computed instantly (Fig. 3a). We subsequently call the computed trajectories the *skeleton trajectories*, and the intersection points the *nodes*.

Lattice initialization: For each new set of PSLs and nodes along them, the globally conforming lattice is computed in turn. Firstly, each node is used as center point for a quadrilateral element, the so-called *lattice seed element*. The ratio of the edge lengths of seed elements is according to the ratio of the principal stresses (Fig. 3a), its edges are along PSLs. A global scale factor controls the size of elements. The elements' edges, per construction, lay on stress trajectories, and we consider them in the upcoming stages to construct a conforming lattice.

Constrained optimization: Given the lattice seed elements, with their nodes being connected via the skeleton trajectories, we aim at connecting the edges of connected elements via stress trajectories. Edges along the major or minor stress direction, respectively, should be connected via a major or minor stress trajectory. However, the edges of connected elements lay on different stress trajectories in general (as shown in Fig. 3b). Thus, we pose an optimization problem with the objective to scale the seed elements individually so that corresponding edges of connected elements lay (approximately) on the same PSL (Fig. 3c). We call the scaled seed elements conforming.

Beam construction: From the conforming seed elements we construct so-called *beams*, which connect the elements along the skeleton trajectories. Beams are sequences of lattice elements, starting and ending in a seed element, and having center points on the connecting skeleton trajectory. The beam elements vary linearly in size and edge ratio along the connecting trajectories (Fig. 3d). The lengths of the beam elements along this trajectory are relaxed to obtain a smooth transition between the seed elements. Notably, due to this construction the shapes of the beam elements cannot indicate any non-linear behavior of the stress ratio along a beam.

In the conforming lattice, the beams follow the principal stress directions only approximately. In extreme cases, the optimization might even suggest element shapes that do not accurately represent the local stress state. By an additional coloring, such cases can be emphasized, hinting towards regions where the lattice needs to be refined. In the following, we describe the major components of the construction process for generating a conforming lattice. We focus on generating such a lattice for a given 2D stress field, and provide details for the handling of 3D stress fields later on.

4. Element construction

Our approach aims at encoding the stress anisotropy into the shape of lattice elements, i.e., the ratio of the edge lengths along the major (l_1) and minor (l_2) principal directions should be equal to σ_1/σ_2. Because of the intrinsic divergence/convergence of PSLs, however, it's not feasible, in general, to strictly ensure this ratio for all lattice elements. Thus, we explicitly enforce this constraint only at the lattice seed elements.

4.1. Lattice seed elements

The lengths of the edges of seed elements are determined as

$$\begin{aligned} l_1 &= l_{max} \cdot \sqrt[p]{|\sigma_1| / \max(|\sigma_1|, |\sigma_2|)} \\ l_2 &= l_{max} \cdot \sqrt[p]{|\sigma_2| / \max(|\sigma_1|, |\sigma_2|)}, \end{aligned} \tag{2}$$

where we prescribe the permitted maximum length l_{max}, as well as a penalty term p that reduces the effect of local extreme values on the construction process. We set $p = 2$ in our examples. Lengths below a prescribed length l_{min} are clamped to this value.

To create the lattice seed element with the computed edge lengths for a given node, we create two points on each the major and the minor PSL going through that node (Fig. 4a). From the intersection of the minor and major PSL, respectively, going through these points, the element corners are determined.

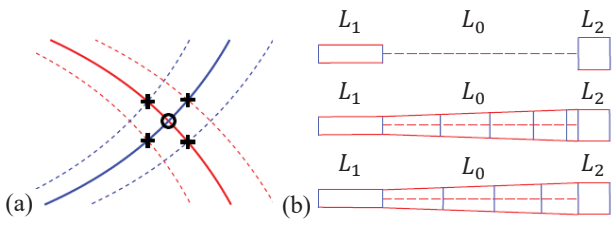

Figure 4: *(a) Lattice seed element construction. (b) From top to bottom, the two seed elements, the non-fitting beam elements, and the fitting elements improved by Eqn. 3.*

4.2. Beam elements

Beams connecting the seed elements along the trajectory skeleton are created by inserting new lattice elements along the skeleton via a relaxation process. Therefore, let us assume that the seed elements have been scaled properly (via the optimization described in Sec. 5, so that the edges of connected elements lay on the same PSL. The construction process considers the length of the PSL from one node to the other one, as well as the lengths of the edges of the lattice seed elements corresponding to these nodes (Fig. 4b).

According to Fig. 4b, let us assume that L_1 and L_2, respectively, refer to the length of the longer and shorter element, and the construction proceeds from L_1 to L_2. L_0 is the distance between the two nodes along the skeleton trajectory. Then, the number of new elements is the number of times an element with average length $l = 0.5 \cdot (L_1 + L_2)$ can be placed between the two nodes, i.e., $N_l = round(l_0/l)$. If $N_l - 1 > 0$, the size l_i of the $i-th$, $i = 1 : N_l$ element is given by

$$l_i = \beta_1 + \beta_2 \cdot (N_l - i), \tag{3}$$

where $\beta_1 = (L_0 - (L_1 - L_2))/N_l$ and $\beta_2 = (L_1 - L_2)/\sum_{k=1}^{N_l-1} k$.

Since it is not always possible to fill exactly the distance between two nodes with new elements, the remaining portion is distributed equally to all elements along the connection. The length of the element edges along the respective other principal stress direction is linearly interpolated between the corresponding values at the nodes.

5. The conforming lattice structure

The seed elements have to be scaled properly so that the edges of connected elements lay on the same PSL. We propose a constrained optimization process and compute the optimized seed elements via a gradient-based optimizer, so that the matching is achieved as good as possible.

5.1. The constrained optimization problem

The optimization problem can be described as

$$\begin{aligned} &\min_x \quad f(x), \\ &\text{s.t.} \quad g(x) \leqslant 0, \quad \textit{and} \quad l_{\min} < x < \lambda l_{\max} \end{aligned} \tag{4}$$

Here, x is the design variable, $f(x)$ is the objective function that measures the un-matching situation, and $g(x)$ is the constraint function that keeps the changes of seed elements within a permitted range. λ is a magnification factor to expand the design domain of the design variables. It is set to 1.5 in our experiments.

Design variables As mentioned in Sec. 4.1, the seed elements are constructed by placing 4 new points around each node, and using the major and minor PSLs through these points to determine the element corners. We take the distances of the new points to the node as the design variables (Fig. 5).

The 4 design variables at each node are the local design variables, and the 4 PSLs traced from each of these nodes are the local

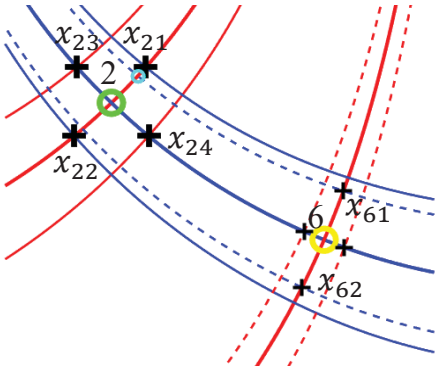

Figure 5: *Schematic illustrating of the optimization process for nodes 2 (green) and 6 (yellow). Thin solid and dashed lines are the support trajectories, crosses refer to the design variables. The first concomitant design variable of x_{21} is x_{61}.*

support trajectories. Let us also introduce the concomitant design variables and support trajectories. By this, we mean the local design variables and PSLs from the nodes that are connected to the current node (Fig. 5).

Constraints The optimization constraints are used to ensure that the seed elements still encode the stress ratio in their shape. According to Fig. 5, the ratio of the i-th seed element is given by

$$r_i = \frac{x_{i1} + x_{i2}}{x_{i3} + x_{i4}}, \tag{5}$$

Furthermore, Eqn. 6 is used to make the change of the seed elements not exceed a predefined range

$$(\frac{r_i - r_i^*}{r_i^*})^2 \leqslant \xi^2. \tag{6}$$

Here, r_i^* refers to the initial shape of the element, and ξ is the permitted change of the element's shape. From this, we obtain the constraint function $g_i(x)$ as

$$g_i(x) = (\frac{1}{r_i^*}\frac{x_{i1} + x_{i2}}{x_{i3} + x_{i4}} - 1)^2 - \xi^2 \leqslant 0, \quad i = 1 : m, \tag{7}$$

where m is the number of nodes. The value of ξ has a significant effect on the convergence of the optimization process and the generated results. A small ξ ensures a limited change to the shape of seed lattice elements. However, it necessitates more iterations in the optimization process and may even lead to non-convergent results.

5.2. Objective function

The objective function is designed to measure how far the current lattice structure is from conformity. To characterize the conformity, we measure the deviation between the local support trajectory and the corresponding concomitant support trajectories. Considering x_{ij}, the j-th design variable of the i-th node, and assuming it has K_{ij} concomitant support trajectories, then the conformity $f_{ij}(x)$ for x_{ij} is defined as

$$f_{ij}(x) = \sum_{k=1}^{K_{ij}} (x_{ij} - \hat{x}_{ijk})^2, \tag{8}$$

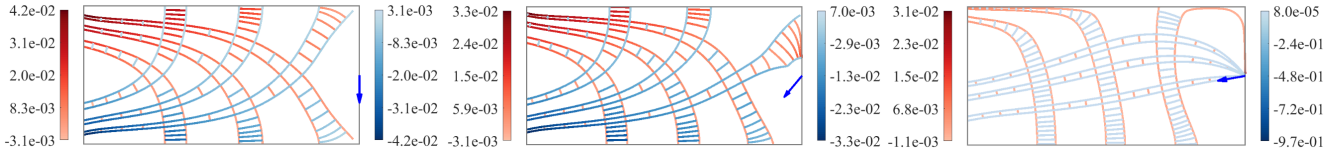

Figure 6: *Coherence of conforming lattices that are generated automatically from the leftmost lattice when different load conditions (indicated by blue arrows) are applied to the 2D cantilever (Fig. 7a).*

where $\hat{x}_{ijk}$ is the distance between the i-th node and the intersection of the k-th concomitant support trajectory with the PSL through x_{ij}. The global conformity $f(x)$, considering m nodes with all 4 design variables, is then given by

$$f(x) = \sum_{i=1}^{m} \sum_{j=1}^{4} \sum_{k=1}^{K_{ij}} (x_{ij} - \hat{x}_{ijk})^2 \tag{9}$$

By substituting Eqns. 7 and 9 into Eqn. 4, we obtain the optimization problem.

Solving the optimization problem requires to trace PSLs and compute their intersections in each iteration step. Therefore, we relax $\hat{x}_{ijk}$ to efficiently find an approximation without computing PSLs. This relaxation builds upon our observation that the design variables are mutually dependent via the concomitant design variables. Thus, the objective function becomes an explicit function with respect to the design variables, if the approximation $h_{ijk}(x)$ of $\hat{x}_{ijk}$ is explicit with respect to x.

To construct $h_{ijk}(x)$, we use polynomial interpolation of samples that are obtained by moving the concomitant design variable in a limited range and computing the induced corrections. Then, the coefficients of the polynomial are computed by solving

$$\begin{bmatrix} X_1^{N-1} & X_1^{N-2} & \cdots & X_1^0 \\ X_2^{N-1} & X_2^{N-2} & \cdots & X_2^0 \\ \vdots & \vdots & \ddots & \vdots \\ X_N^{N-1} & X_N^{N-2} & \cdots & X_N^0 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_N \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{bmatrix} \tag{10}$$

Here, p_s, $s = 1:N$ are the coefficients of the polynomial, N is the number of samples, and X_i and Y_i, $i = 1:N$ are the sampled concomitant design variables and their corresponding corrections. The variables $\hat{x}_{ijk}$ can now be approximated via

$$\hat{x}_{ijk} \approx h_{ijk}(x) = \sum_{s=1}^{N} p_s \cdot x^{N-s}. \tag{11}$$

N is set to 3 in our experiments. By substituting Eqn. 11 into Eqn. 9, the explicit expression of the objective function with respect to the design variable is available, and the final optimization equation becomes

$$\begin{aligned} &\min_x \; f(x) = \sum_{i=1}^{m} \sum_{j=1}^{4} \sum_{k=1}^{K_{ij}} (x_{ij} - h_{ijk}(x))^2, \\ &\text{s.t. } g_i(x) = (\tfrac{1}{r_i^*} \tfrac{x_{i1}+x_{i2}}{x_{i3}+x_{i4}} - 1)^2 - \xi^2 \leqslant 0, \; i = 1:m, \\ &\quad l_{\min} < x < \lambda l_{\max} \end{aligned} \tag{12}$$

This function, since its derivatives can be computed analytically, gives rise to efficient gradient-based optimizers, like the method of moving asymptotes (MMA) [Sva87] that is used in our work.

5.3. Coherent visualization of stress changes

In many practical applications, domain experts are interested in analysing the variations in the internal stress state due to changes in the external load conditions or when a different material is simulated. To enable an effective comparison of different stress states, their visualizations should not change significantly when only marginal changes have occurred.

With our method this is difficult to achieve, because the user specifies only few initial seed nodes, while the other nodes are computed automatically from the intersections of the traced PSLs. When the seed nodes are fixed and used for generating the conforming lattice for a new stress state, due to changes in the PSLs the other nodes might be at vastly different locations than they occurred in the previous lattice. On the other hand, if the initial nodes were only slightly moved in the new field, the resulting lattice structure might be very similar to the initial one.

To preserve positional coherence of all nodes—and thus structural coherence of the lattice structures—for varying stress states, we introduce a globally optimal placement scheme that attempts to minimize the summed positional changes over all nodes when generating a conforming lattice for a new stress distribution. Therefore, we establish the following equation for the placement optimization:

$$\begin{aligned} &\min \longrightarrow \quad \sum_{i=1}^{m} |p_i - \hat{p}_i|^2, \\ &\text{s.t.} \quad p_j \in \Gamma_j, \; j = 1:\hat{m} \end{aligned} \tag{13}$$

Here, $\hat{p}_i$ and p_i, $i = 1:m$, respectively, are the node positions in the previous and current lattice. p_j, $j = 1:\hat{m}$ are the seed nodes, which are the design variables in the optimization. Γ_j, $j = 1:\hat{m}$ are the corresponding design domains, which restrict the movements

of the seed nodes. Thus, the locations of nodes that are created automatically are regularized by these nodes. Since it is not possible, in general, to derive a differentiable formulation of the automatically computed locations of new points with respect to the seed points, a gradient-based optimizer cannot be used. Instead, we resort to the derivative-free optimizer CMA-ES proposed in [Han16], which only requires iterative evaluations of the free node positions for varying seed positions. I.e., in every iteration the seed nodes are slightly moved so that the positional changes of all nodes are optimized.

As an example, consider the 2D cantilever in Fig. 7a, to which a concentrated force is applied at the point P_3. The initial load direction is downward, and it is then changed continually in 9 steps of 20 degrees clockwise, so that eventually the load direction turns to 180 degrees and the structure is under an upward force. We input three seed nodes p_1, p_2 and p_3, with initial positions (125,125), (250,125) and (375,125), respectively. The corresponding design domains Γ_i, $i = 1:3$ of p_i are indicated by circles around each p_i.

When comparing the lattice structures from different simulation steps in Fig. 6, it can be seen that number of nodes and skeleton trajectories do not change, the lattice slightly rotates clockwise, and the region where the structure is under compression becomes increasingly larger from left to right (see the shape change of the lattice elements). In particular, even though significant changes in the stress field occur, the conforming lattices do not change their topology and geometric changes are tried to be minimized. This gives rise to an effective comparison of all fields.

6. Results and analysis

We use the solid structures in Fig. 7 to validate our method. Stress fields are computed by finite hexahedral element analysis, with the Young's modulus and Poisson ratio of all solids set to 1 and 0.3, respectively. The second solid in Fig. 7 is obtained from the first one by inserting holes.

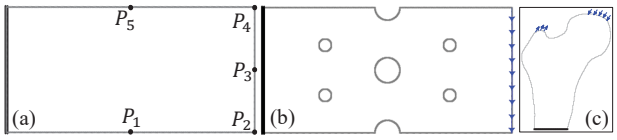

Figure 7: *The 2D solid objects used in our experiments. Blue arrows indicate the load conditions. (a) Cantilever with fixed left edge, discretized into* 250×500 *finite elements. Points* $P_1, \ldots, P_5$ *indicate the positions were different loads are applied. (b) Perforated plate, adapted from the cantilever. (c) Slice through a 3D CT-scan of a femur, fixed at the bottom and discretized into* 182×140 *finite elements.*

All experiments are carried out using the *MATLAB R2019a - academic use* release, on a workstation running Windows 10 and equipped with 8 cores (Intel Xeon W-2123, @3.60Ghz) and 64GB RAM. The processing times range from roughly 5 seconds for the 2D examples up to almost 25 seconds for the 3D examples discussed below. Since the number of constraint functions in Eqn. 12 equals the number of nodes that are used to construct the skeleton trajectories (i.e., Sec. 5), processing times strongly depend on this number. For large numbers, also the probability to get stuck in local optima increases.

To validate the capability of our method to represent complicated stress scenarios, we simulate the stress distribution in the 2D cantilever (Fig. 7a) using three different load conditions: The 1st test case is obtained by applying a rightward concentrated force and a downward force on P_3 and P_1, respectively. The 2nd case is generated by applying two rightward concentrated forces on P_3 and P_5 separately. The 3rd case is obtained by applying a downward distributed force on the edge P_2P_4. The 1st and 2nd test cases are used to demonstrate the construction of a conforming lattice when a trisector and a wedge degenerate point, respectively, exist. Via the 3rd case we shed light on a scenario where a large number of nodes is initially specified. In all test cases, we set $l_{min} = 2.5\delta$ and $l_{max} = 4l_{min}$. In Eqn. 12, $\xi = 0.3$ is used to restrict the change of the seed lattice shape during the optimization, here setting it 0.3 is a consideration from both the convergence and effectiveness after observing a series numerical examples.

For all three test cases, the support trajectories before and after optimization are shown in Fig. 8, In a pre-process, degenerate points are computed and shown together with the corresponding topological skeleton. The topological skeleton divides the domain into sub-domains where the corresponding PSLs have similar behavior [DH94]. In particular, this information is used to avoid placing seed nodes too close to critical points or topological skeletons, so that the computed beam structures overlap multiple topological regions.

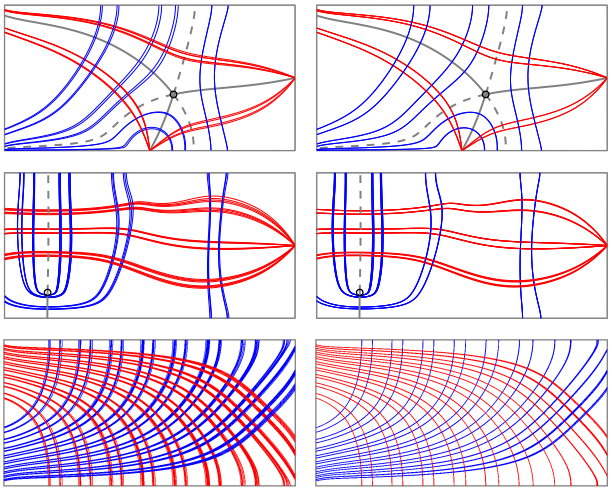

Figure 8: *Top to bottom: Support trajectories for the 2D cantilever under different loads before (left) and after (right) optimization, including degenerate points (small circles), and the topological skeleton with PSLs along the major (grey solid lines) and minor (dashed lines) principal directions.*

Notably, the optimization fails if the support trajectories fall within different topological regions (Fig. 9). In this case, either the

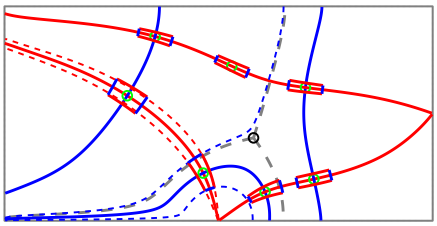

Figure 9: *Failure case. Support trajectories (blue dashed lines) enclosing a degenerate point (circle). Topological skeleton in grey.*

size of the lattice seed elements needs to be scaled down or the selected node needs to be relocated.

The conforming lattices for the three test cases are shown in Fig. 10 and Fig. 1(left). The values of σ_1 and σ_2 along the major and minor principal directions are encoded in shades of red and blue. Visualization of stress anisotropy are shown in the Appendix.

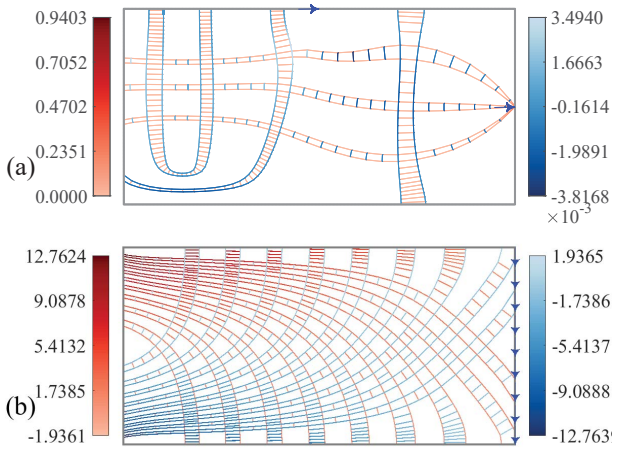

Figure 10: *Visualization of stresses in the cantilever under different loads using conforming lattices. Blue arrows indicate the loads corresponding to our 2nd (top) and 3rd (bottom) test case.*

The perforated plate (Fig. 7b) is used as an additional test case to validate our method. The stress field is simulated by applying a distributed force acting downward on the right boundary of the plate. The optimized support trajectories are shown in the top of Fig. 11, with color coding according to major and minor stress direction (left) and according to the von Mises stress (right) to show the potential stress concentration. In the bottom of Fig. 11, the same visualizations are shown for the 2D femur slice. The loads mimic compression due to body weight (downward acting loads) and tension due to muscle forces (upward acting loads). In Fig. 1(middle) and Fig. 1(right), major and minor principal stresses along the PSLs in both datasets are encoded into shades of red and blue.

Convergence A conforming lattice is constructed by a constrained optimization. The convergence behavior of both the objective function and the constraint function for all five test cases are collectively shown in Fig. 12a and b. Here, the normalized objective

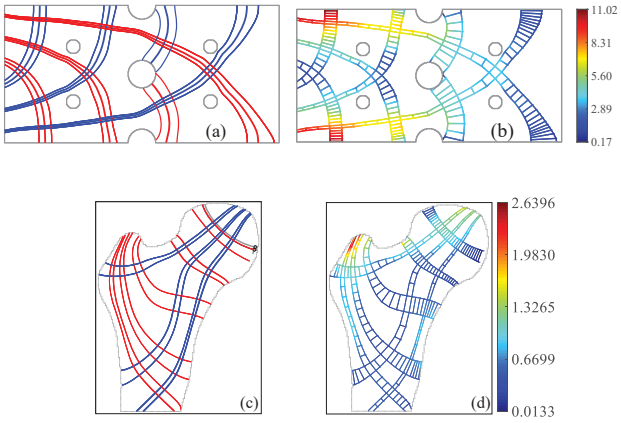

Figure 11: *Perforated plate (top) and 2D femur slice (bottom). Optimized support trajectories (left), and conforming lattice with von Mises stress encoded along principal directions (right). In the femur stress field, PSLs cannot reach the boundary in the right upper region, because of the two degenerate points (black circles).*

function $f^*(x) = f(x)/max(f(x))$ is used. The convergence plots indicate that the optimization always converges after less than 50 iterations. We further analyse the accuracy by which the edges of beam elements follow the PSLs. Therefore, we measure the average directional deviation between the element edges at the element corner points and the exact direction of the PSLs. This deviation is always below 4 degrees, yet we also observed outliers of more than 20 degrees in highly diverging situations. Fig. 12c, b shows the seed elements before and after optimization, superimposed for comparison, for the 2nd and 3rd test case, respectively. A numerical analysis verifies that all changes in aspect ratio are within the prescribed tolerance, $\xi = 0.3$, which confirms convergence of the optimization process.

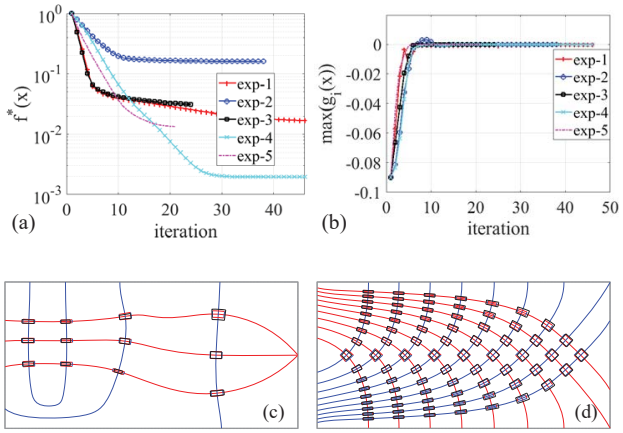

Figure 12: *Top: Convergence analysis for all 2D experiments. (a) Normalized objective function. (b) Constraint function. Bottom: Comparisons of seed elements before and after optimization, for the 2nd (c) and 3rd (d) test case. Elements with colored edges depict the initial seed elements, black edges indicate optimized elements.*

7. 3D lattice structure

In principle, the conforming lattice in a 3D stress tensor field can be computed in much the same way as in 2D. However, since the PSLs in a 3D tensor field do not intersect in general, our proposed optimization process is not applicable. Thus, in 3D we waive this process and only make use of the proposed beam construction process. Conformity is ensured by progressively growing beams along the PSLs, thereby adjusting the cells' extents to convey the local stress state.

7.1. Beam growth

The seed element in 3D is constructed so that its edges are aligned with the three principal stress directions and the edge lengths are according to the ratio of the principal stresses. Since the edges do not necessarily coincide with PSLs, we place points at the edge midpoints, compute the PSLs corresponding to the edge direction, and relax the cell corner points towards these PSLs. This process is repeated iteratively until the changes are below a given tolerance.

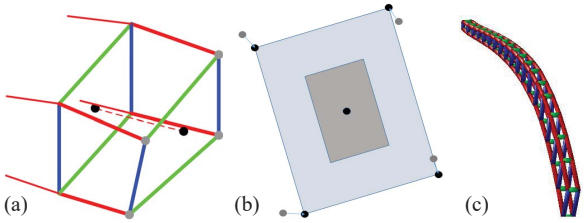

Figure 13: *Illustration of the element construction process. (a) Tracing along PSLs, (b) Fitting of element face. (c) The final beam structure in 3D.*

For every face of the seed element and starting from the face's corner points, we trace out 4 new points along the PSLs "orthogonal" to that face. The step size along the PSLs is the same as the length of the seed element along that direction (Fig. 13a). When connecting the new points (grey dots in Fig. 13b) to form a face of the new element, however, the edges of that face do not coincide with PSLs in general. Thus, we also trace out a 5th point along the skeleton trajectory of the current beam, using the same step size as before (black dot in Fig. 13a). We then center a planar quadrilateral at that point, with its edges aligned along the respective other PSLs at this point, and the edge lengths according to the stress ratio along these PSLs (dark grey quadrilateral in Fig. 13b). The newly generated points are projected onto the plane of the quadrilateral (grey dots in Fig. 13b). Since we know which corner point of the quadrilateral corresponds to which new point, we now scale the quadrilateral isotropically so that the sum of the distances between it's corner points and their corresponding new points is minimized (light grey quadrilateral in Fig. 13b). The new points are then snapped to their corresponding corner points to form the final face. Fig. 13c shows an entire beam that is generated as described.

It is clear that the edges of the beam elements do not follow the PSLs exactly, since they are aligned solely with the PSLs at the selected center point. Apart from cases where trajectories are highly diverging, however, the elements align fairly well with the PSLs in all of our experiments. In particular, the average directional deviation between the element edges at the element corner points and the exact direction of the PSLs was always below 5 degrees. An extraordinary case occurs when two trajectories have been snapped to each other. Then, the elements at the start and end points are generated along different beams, and they do not consider the stress variation along the connection. We handle this case by interpolating the beam elements linearly along the connection, in the same principal way as described for the 2D setting.

7.2. Visual mapping

To visualize the generated 3D beam structure, all element edges are rendered as tubes for improved visibility. In addition, the stress magnitudes along the major, medium, and minor PSLs are mapped to red, green, and blue, respectively, with the magnitudes from lowest to highest encoded by increasing saturation. To reduce visual clutter, beam elements are rendered as opaque cubes, and the strength of anisotropy of the stress magnitudes is encoded into greyscales of the element faces. Here, the following coloring scheme is used to compute the intensity γ_{3D}:

$$\gamma_{3D} = \ln\left(\frac{\max(|\sigma_1|,|\sigma_3|)}{\min(|\sigma_1|,|\sigma_3|)}\right) \tag{14}$$

7.3. Test cases

In our experiments we consider two solid objects (Fig. 14), a 3D cantilever and a human femur. In the first two experiments using the 3D cantilever, we apply a distributed force to face $P_1P_2P_3P_4$, which is then exchanged by a distributed force exerting on edge P_1P_2 to introduce a torque.

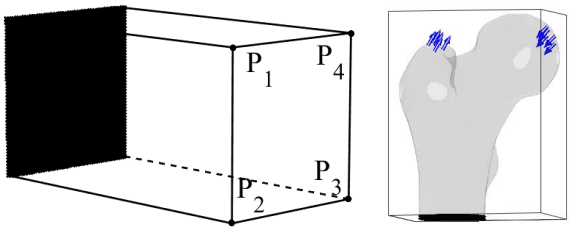

Figure 14: *The 3D solid objects used in our experiments. Blue arrows indicate the load conditions, regions where the objects are fixed are shown in black. Left: 3D cantilever with fixed left face, discretized into* $50 \times 100 \times 50$ *finite elements. Different loads are applied at points* $P_1,\ldots,P_4$*. Right: Femur fixed at the bottom, discretized into* $140 \times 92 \times 182$ *finite elements.*

In Fig. 15a, b, beam structures in the cantilever stress fields are shown when only one single beam is traced along the PSLs at the initial seed point. Most importantly, the 3D beam structures can simultaneously visualize the principal stress directions, the divergence and convergence of PSLs, as well as the anisotropy of the stress magnitudes. Notably, the torque that is introduced by the load in the second experiment is clearly conveyed by the torsion of the beams along the major and minor stress directions. Visually,

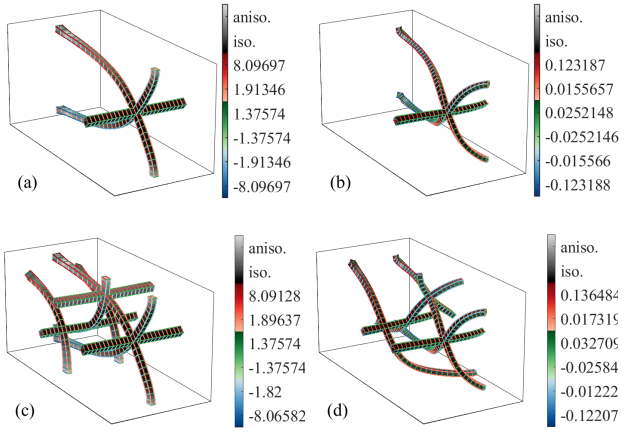

Figure 15: *Beam structures showing the stresses in the 3D cantilever under different load conditions. Shades of grey characterize the strength of anisotropy of the beam elements. Top: One single seed element is selected. Bottom: Beam structures are extended by growing beams from additional elements.*

the single beams look similar to hyperstreamlines, yet there are essential differences between them. The most important are that the beams encode (relative) local and global information concerning all three major stress directions, and can be stitched together in a conforming way to better represent the 3D stress distribution. The latter is demonstrated in Fig. 15c, d, where additional beam elements are selected and new branches are grown along them.

We perform one last experiment using the 3D femur (Fig. 14right), where multiple seeds are input sequentially in order to obtain a complete image of the stress distribution. As seen in Fig. 16, faces colored in light greyscale also appear in some middle parts of the beams, i.e., high anisotropy does not only occur in regions close to where the loads are applied, but also in the interior. Region of high compression due to body weight and high tension due to muscle forces can be conveyed effectively in the upper right and left part of the femur, respectively. In addition, the twisting of beams indicate that the stress field is under severe torsion. It is worth noting that these effects are difficult to convey via alternative techniques like hyperstreamlines or by drawing single PSLs along any of the major stress directions.

8. Conclusion and outlook

We have introduced a novel method to visualize 2D stress tensor fields via a conforming lattice that follows the PSLs, conveys their divergent/convergent behaviour, and encodes stress type and relative stress magnitude. The method is global in that it allows following the paths along which stresses are transmitted through the domain. The construction of a 2D conforming lattice is formulated as an optimization problem, which adjusts the lattice elements so that they conform to the local stress state. We have shown a modification to visualize 3D stress fields, by progressively building a beam structure comprised of hexahedral elements using a sequential growth process.

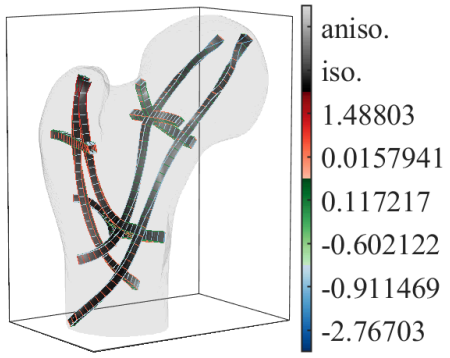

Figure 16: *Visualization of stresses in the 3D femur using beams that are grown progressively from user selected seed elements.*

In the future, we aim to extend the method to generate a space-filling stress-following grid in 2D and 3D. In 2D, we will further investigate construction processes that consider sharing of support trajectories between beams, in combination with seeding strategies that can automatically distribute beams so that they densely cover the domain. In 3D, since a globally conforming structure does not exist in general, we will investigate relaxation schemes to compute a pseudo-conforming yet dense grid structure. Therefore, it will be interesting to look into hex-meshing approaches for arbitrary geometries, and to adapt them to the specific needs. Since occlusions become paramount already in the case of rather sparse 3D spatial grid structures, we will further investigate dedicated visualization techniques for such structures, e.g., by considering focus+context techniques including feature-based element highlighting.

9. Acknowledgements

This work was partially funded by the German Research Foundation (DFG) under grant number WE 2754/10-1 "Stress Visualization via Force-Induced Material Growth".

10. Appendix

We provide additional visualizations of the anisotropy of minor and major stresses in the 2D cantilever under different loads (Fig. 10). Fig. 17 shows the anisotropy measured by $\gamma_{2D} = \ln(\frac{|\sigma_1|}{|\sigma_2|})$. By comparing the anisotropy with the aspect ratio of beam elements in Fig. 10, good agreement of the aspect ratio of elements and the anisotropy can be observed.

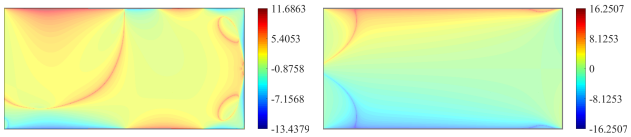

Figure 17: *Anisotropy of local stresses in the cantilever under different loads (2nd (left) and 3rd (right) test case).*

References

[ACSD*03] ALLIEZ P., COHEN-STEINER D., DEVILLERS O., LÉVY B., DESBRUN M.: Anisotropic polygonal remeshing. *ACM Trans. Graph. 22*, 3 (July 2003), 485–493. 3

[BES15] BUSSLER M., ERTL T., SADLO F.: Photoelasticity raycasting. In *Computer Graphics Forum* (2015), vol. 34, Wiley Online Library, pp. 141–150. 2

[DGBW09] DICK C., GEORGII J., BURGKART R., WESTERMANN R.: Stress tensor field visualization for implant planning in orthopedics. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 1399–1406. 2, 3

[DH93] DELMARCELLE T., HESSELINK L.: Visualizing second-order tensor fields with hyperstreamlines. *IEEE Computer Graphics and Applications 13*, 4 (1993), 25–33. 3

[DH94] DELMARCELLE T., HESSELINK L.: The topology of symmetric, second-order tensor fields. In *Proceedings of the conference on Visualization'94* (1994), IEEE Computer Society Press, pp. 140–147. 3, 7

[Fro48] FROCHT M. M.: *Photoelasticity*, vol. 2. J. Wiley, 1948. 2

[GRT17] GERRITS T., RÖSSL C., THEISEL H.: Glyphs for space-time jacobians of time-dependent vector fields. 2

[Han16] HANSEN N.: The cma evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772* (2016). 7

[HFH*06] HOTZ I., FENG L., HAGEN H., HAMANN B., JOY K.: Tensor field visualization using a metric interpretation. In *Visualization and processing of tensor fields*. Springer, 2006, pp. 269–281. 3

[HLL97] HESSELINK L., LEVY Y., LAVIN Y.: The topology of symmetric, second-order 3d tensor fields. *IEEE Transactions on Visualization and Computer Graphics 3*, 1 (1997), 1–11. 3

[HSH07] HLAWITSCHKA M., SCHEUERMANN G., HAMANN B.: Interactive glyph placement for tensor fields. In *International Symposium on Visual Computing* (2007), Springer, pp. 331–340. 2

[KASH13] KRATZ A., AUER C., STOMMEL M., HOTZ I.: Visualization and analysis of second-order tensors: Moving beyond the symmetric positive-definite case. In *Computer Graphics Forum* (2013), vol. 32, Wiley Online Library, pp. 49–74. 2

[Kin04] KINDLMANN G.: Superquadric tensor glyphs. In *Proceedings of the Sixth Joint Eurographics-IEEE TCVG conference on Visualization* (2004), Eurographics Association, pp. 147–154. 2

[KMH11] KRATZ A., MEYER B., HOTZ I.: A visual approach to analysis of stress tensor fields. In *Dagstuhl Follow-Ups* (2011), vol. 2, Schloss Dagstuhl-Leibniz-Zentrum für Informatik. 2

[KW06] KINDLMANN G., WESTIN C.-F.: Diffusion tensor visualization with glyph packing. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006), 1329–1336. 2

[KWS*08] KINDLMANN G., WHALEN S., SUAREZ R., GOLBY A., WESTIN C.: Quantification of white matter fiber orientation at tumor margins with diffusion tensor invariant gradients. In *Proc. Intl. Soc. Mag. Reson. Med* (2008), vol. 16, p. 429. 2

[RBN*19] RAITH F., BLECHA C., NAGEL T., PARISIO F., KOLDITZ O., GÜNTHER F., STOMMEL M., SCHEUERMANN G.: Tensor field visualization using fiber surfaces of invariant space. *IEEE Transactions on Visualization and Computer Graphics 25*, 1 (Jan 2019), 1122–1131. 3

[RKZZ18] ROY L., KUMAR P., ZHANG Y., ZHANG E.: Robust and fast extraction of 3d symmetric tensor field topology. *IEEE transactions on visualization and computer graphics 25*, 1 (2018), 1102–1111. 3

[SK10] SCHULTZ T., KINDLMANN G. L.: Superquadric glyphs for symmetric second-order tensors. *IEEE transactions on visualization and computer graphics 16*, 6 (2010), 1595–1604. 2

[SK16] SELTZER N., KINDLMANN G.: Glyphs for asymmetric second-order 2d tensors. In *Computer Graphics Forum* (2016), vol. 35, Wiley Online Library, pp. 141–150. 2

[Sva87] SVANBERG K.: The method of moving asymptotes—a new method for structural optimization. *International journal for numerical methods in engineering 24*, 2 (1987), 359–373. 6

[Tim83] TIMOSHENKO S. S.: *History of Strength of Materials*. Dover, 1983. 2

[WB05] WILSON A., BRANNON R.: Exploring 2d tensor fields using stress nets. In *VIS 05. IEEE Visualization, 2005.* (Oct 2005), pp. 11–18. 3

[ZP04] ZHENG X., PANG A.: Topological lines in 3d tensor fields. In *Proceedings of the conference on Visualization'04* (2004), IEEE Computer Society, pp. 313–320. 3

[ZP05] ZHENG X., PANG A.: 2d asymmetric tensor analysis. In *VIS 05. IEEE Visualization, 2005.* (2005), IEEE, pp. 3–10. 2

[ZS18] ZOBEL V., SCHEUERMANN G.: Extremal curves and surfaces in symmetric tensor fields. *The Visual Computer 34*, 10 (Oct 2018), 1427–1442. 3

[ZYLL08] ZHANG E., YEH H., LIN Z., LARAMEE R. S.: Asymmetric tensor analysis for flow visualization. *IEEE Transactions on Visualization and Computer Graphics 15*, 1 (2008), 106–122. 2