Geodesic Composite Fiber Generation Algorithm

3D Model - STL or X3D

- Steps to generate x3d example:

1. Open blender. Middle mouse button zooms. Alt left mouse button rotates. Shift alt left mouse button slides.
2. Delete default cube.
3. Import .x3d file and zoom into it.
4. Alt-right-mouse button to select it (may freeze for a minute... just be patient), then switch from object mode to edit mode.
5. Go into edge selection mode. Select seams. Mark seams.
6. Go into face selection mode. Select all faces.
7. Perform unwrapping. Unwrap params should appear at the bottom of the toolbar to your left.
    1. Adjust parameterization params from "angle based" to "conformal mode" (bottom of tool shelf (press ‘T’))
8. In lower window, select UV/Image Editor.
9. Verify parameterization.
    1. MAKE SURE HIGH PRECISION TEXTURE COORDINATE MAPPING IS ENABLED... see blender_io_scene_x3d.patch (also in software/blender_x3d_hirestex...) must be turned on in preferences...addons.
10. Adjust resolution when exporting suggest 8 digits.
11. Export as X3D to a _UV.x3d file
    1. WARNING: blender will rotate the x,y,z axes on x3d import, but it rotates them back on x3d export

Terminology

- Start point – Beginning point to start the algorithm (origin)
- Fiber start point – The very first point of each fiber geodesic
- Fiber initialization geodesic – A geodesic that computes each fiber start point
- Element – One triangle in the model

Algorithm Inputs

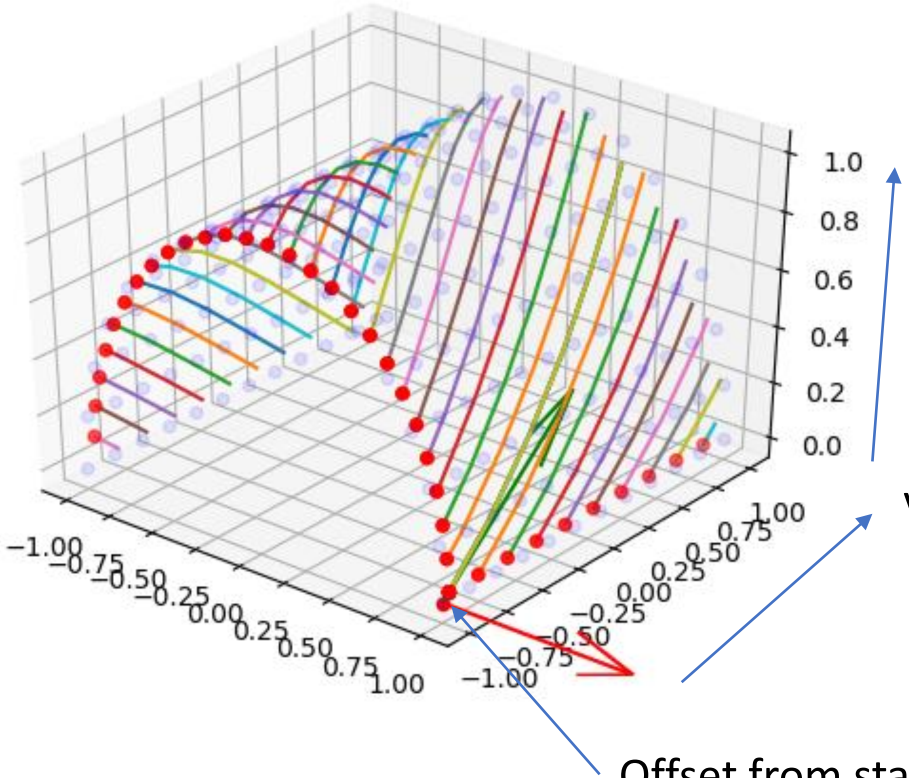
- filepath – path to the model file (x3d or stl (stl will come in future update))
- startpoint – The origin of each fiber initialization geodesic
- fiberangle – Fiber orientation in degrees
- fiberdirection – A vector describing the fiber orientation (relative to the surface and respective of the primary axis directions, x, y, z)
    - Ex. For 30 degree orientation and a primary surface normal of [0, 0, 1]
        - [-cos(fiberangle), sin(fiberangle), 0]
- pointdirectionnormal – a normal that defines the entire surface of interest (to define which surface to look at if on a corner or edge)
- offset – distance to offset from the starting vertex (in case edges aren't straight) (Warning: This will be improved as fiber start point generation is improved)
- fiberint – distance between fiber start points
- E – Youngs Modulus (For strain energy optimization)
- nu – Poisson's ratio (For strain energy optimization)

Algorithm in General

- Compute Geodesics
    - Find fiber start points
        - Send a geodesic in all valid directions
        - Assign fiber start points respecting fiberint
    - Determine surface specific vertices
        - Gathering only the vertices that matter to the surface of interest
            - Decreases amount of data we have to work with
    - For each fiber start point
        - Calculate the geodesic path
    - Find any unassigned vertices and interpolate their uv position

Find fiber start points

- For each principle direction, i.e. [1, 0, 0], [0, 1, 0], etc..
    - Move the starting vertex in the direction of the fibervector with magnitude of “offset”
    - Determine starting element
    - If no starting element is found continue to next principle direction
    - Determine direction and distance in principle direction based on fiberint
    - Rotate fibervector to be in plane with starting element
        - Assign starting elements fiberdirection to be the new fibervector
    - Rotate the fiber start point geodesic direction to be in plane with the starting element
    - Perform the algorithm on slide 8 until an edge is reached
        - While keeping track of each fiber start point, it’s starting element, the adjusted fiberdirection for that element, and it’s (u, v) starting coordinates.



Red point: Fiber start point
Green arrow: Given fiber direction vector
Red Arrow: Point direction normal
Blue dots: Model vertices
Multi-colored lines: Each geodesic

Valid fiber start point geodesic directions

Offset from starting vertex

Calculate Geodesic Path

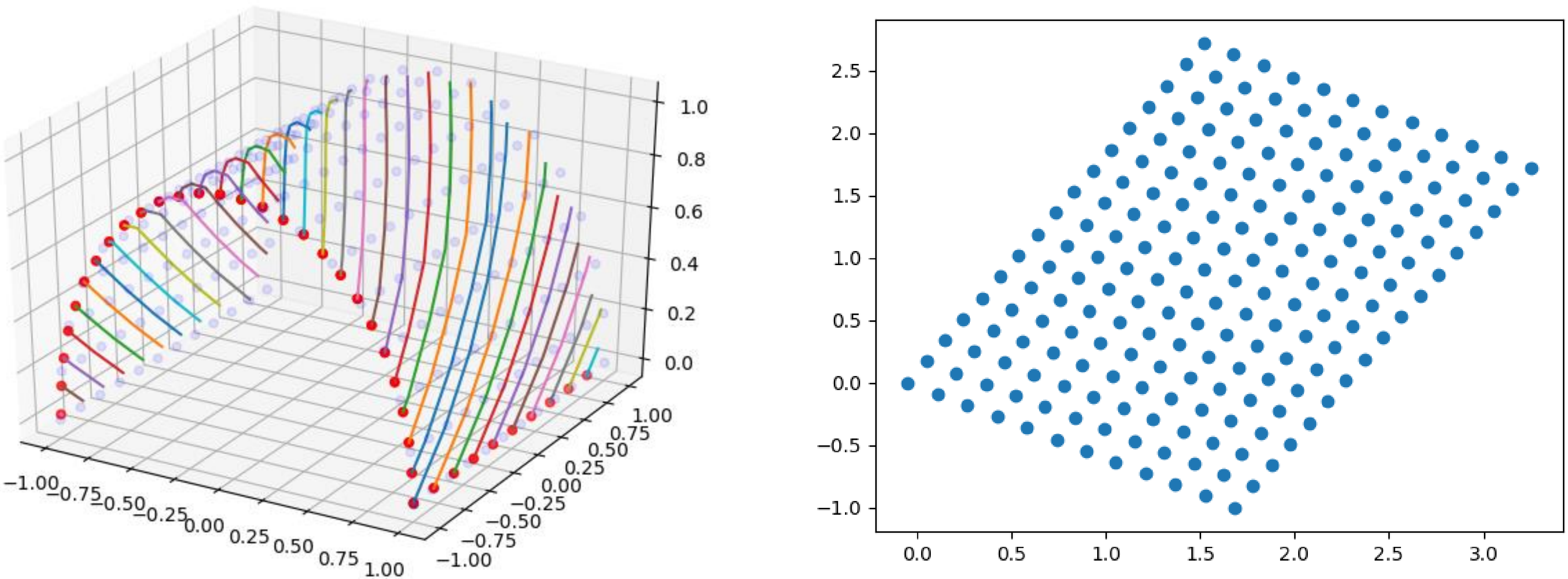
- Given: a start point, start element, and fiber direction
- While access to a valid element and an edge hasn’t been crossed:
    - If
        - The current element is not in list of faces, add it
        - The current element doesn’t have a fiber direction associated to it, add it
    - Calculate the closest vertex from the current point and the u, v distance (See slide 10)
    - If this vertex doesn’t have a u, v assignment OR the current v distance is less than the stored v distance
        - Add the current fiber length, the new u distance, and the u start coordinate of this geodesic
        - Add this v distance and the v start coordinate of this geodesic
        - Replace the current value for this vertex’s uv coordinates with these new uv coordinates
    - Traverse the current element and return the next intersection point, next fiber vector, and the next element (See slide 9)
    - Update the geodesic length given the distance across the current element
    - Update the current point with the new intersection point, the current fiber direction with the new fiber direction, and the current element with the new element

Traversing an Element

- Given: Current element, a point (within this element), and a fiber direction
- Convert the 3D coordinates and fiber direction of the current element into Barycentric coordinates
- Find the intersection point of the fiber direction vector with an edge of the current element
    - This intersection point will be our next point in the geodesic
- Find the neighboring elements
- Determine which element is across the intersection boundary
- Return the next intersection point, the rotate fiber direction for the enxt element, and the next element

Calculating the closest point

- Given: A fiber direction, a start point, an array of points, and a normal
- Calculate the distance from each point to the fiber direction vector that originates from the start point
- Return the parallel distance (u), perpendicular distance (v), and the corresponding 3D point


−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
1.0
0.8
0.6
0.4
0.2
0.0
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0