Radially Expanding Euler Paths for Assembly of Truss Structures

By Harsh G. BHUNDIYA,[1] and Zachary C. CORDERO,[1]

[1] *Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, United States*

Planning the assembly sequence is a critical challenge for the robotic construction of structures, particularly for maintaining stiffness during the construction process. To aid the assembly sequence planning for truss structures, this paper presents an algorithm for sampling Euler paths which expand radially from a starting node. We show that such paths are desirable as they result in intermediate structures with dramatically higher natural frequency than those from randomized assembly sequences. The algorithm is widely applicable to scenarios where intermediate stiffness is required, such as in-space assembly and manufacturing, and motivates future research into other sampling strategies for the assembly of truss structures.

Key Words: Assembly sequence planning, Euler paths, truss structures

1. Introduction

The robotic assembly of truss structures is promising for automating ground-based construction tasks[1] and enabling novel applications such as the in-space assembly and manufacturing of support structures.[2] Some recent approaches to robotic truss assembly include: assembling cellular structures with a mobile robotic platform,[3] assembling pre-fabricated struts with a robotic manipulator,[4] reconfiguring a modular chain,[5] and plastically deforming metal rods into wireframe structures.[6] For all these approaches, one key challenge is planning the assembly sequence, as many exist for the same truss geometry. A truss geometry can be represented as a graph with nodes and edges (Fig. 1), so the assembly sequence corresponds to a path that traverses all the edges, i.e., an Euler path. However, the number of possible Euler paths increases rapidly with graph size,[7] making it difficult to find optimal assembly sequences that maximize important performance metrics. For example, the on-orbit construction of support structures will require maintaining stiffness and surface precision during assembly to ensure structural stability and geometrical accuracy of the final shape.[8]

Given the large design space of assembly sequences for truss structures, previous work has used hierarchical representations of truss sub-assemblies[9] and graph search algorithms to find sequences that optimize specific cost functions.[10] While this approach works well for relatively simple trusses, it scales poorly with structural complexity. Randomized optimization techniques, such as particle swarm optimization[11] and flower pollination,[12] have shown promise for complex assembly sequences, but their computational complexity also increases with graph size.

An alternate approach to assembly sequence planning for complex truss geometries is to develop sampling methods for assembly sequences and understand their performance with relevant metrics. This approach explores the design space and allows identification of heuristics that lead to optimal assembly sequences, and is the approach taken in the present work. In particular, we present an algorithm for sampling radially expanding Euler paths for the assembly of truss structures. The algorithm is based on a classical algorithm for computing Euler paths through an undirected graph, but incorporates a locally optimal criterion for selecting edges that minimize distance to the partially assembled structure. For the application of truss assembly, we show that such paths are desirable as they result in stiffer intermediate structures with higher fundamental natural frequency than other randomized assembly sequences.

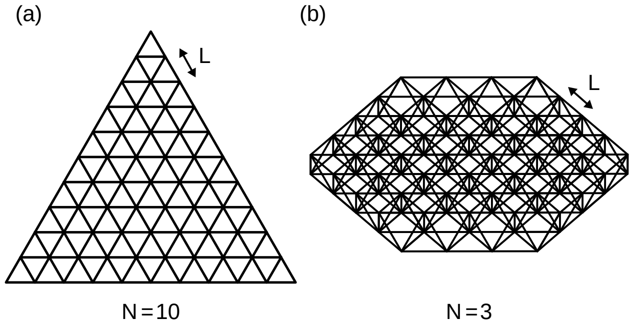

Fig. 1. Geometries of (a) 2D equilateral triangle and (b) 3D tetrahedral truss, parameterized by the strut length *L* and number of units *N*. All struts have equal length.

The remainder of the paper is organized as follows. Section **2.** describes the algorithm for radially expanding Euler paths and the modeling framework for computing the stiffness of intermediate structures. Section **3.** compares the performance of this algorithm with a classical algorithm for computing Euler paths in terms of the fundamental natural frequency. Finally, Section **4.** provides a summary and directions of future work.

2. Methods

2.1. Radially Expanding Euler Paths

The classical method to compute an Euler path through an undirected graph is via the route inspection algorithm.[7] This algorithm has two steps: First, the minimum number of doubled edges are added to the graph until it satisfies Euler's theorem, which holds that there exists an Euler path if and only if there are no more than two nodes with odd degree. Typically, this step is completed by finding a minimum-weight perfect matching that connects the odd degree nodes while minimizing the total length of the added paths.[13] Second, an Euler path is computed through the graph using either the Hierholzer or Fluery algorithms. For a graph with n edges, the Hierholzer algorithm has complexity $O(n)$ and finds closed loops in the graph before concatenating them into one continuous Euler path.[14] Comparatively, the Fluery algorithm has complexity $O(n^2)$ and concatenates neighboring edges into an Euler path without creating bridges. Both these algorithms result in randomized Euler paths that traverse all edges of the graph from a specific starting node.

In this work, we are interested in sampling Euler paths which radially expand from the starting node. For the assembly of truss structures, the intuition is that such paths correspond to sequentially adding struts around the circumference, resulting in greater stiffness during construction. To this end, we modify the Fluery algorithm with a specific criterion for adding edges to the Euler path: only select neighboring nodes that minimize the Euclidean distance to nodes already in the path. This locally optimal criterion selects edges closest to the partially assembled geometry at each step and yields a path that expands radially outward from the starting node, as illustrated in Fig. 2. The pseudocode for this algorithm is summarized in Algorithm 1, with the key selection criterion highlighted in red. As this algorithm is based on the Fluery algorithm, it also has complexity $O(n^2)$.

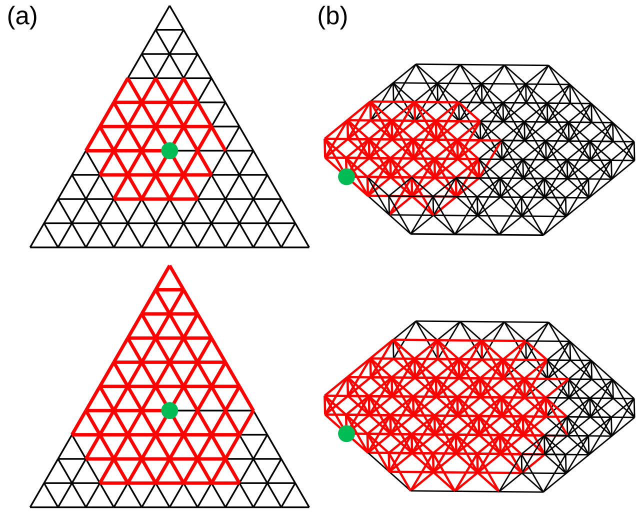

Fig. 2. Radially expanding Euler paths for (a) 2D equilateral triangle and (b) 3D tetrahedral truss that grow from the starting node highlighted in green.

2.2. Finite Element Analysis

To understand the stiffness of intermediate structures that result from radially expanding Euler paths as compared to randomized paths, we use finite element modeling. Specifically, we compute the fundamental free-free natural frequency of various intermediate structures resulting from the Euler path, an important performance metric for the assembly of precise truss structures.[8]

Focusing our analysis to the geometries of Fig. 1, we compute a discrete number of intermediate structures from a specific Euler path and use the finite element software Abaqus to compute their fundamental natural frequencies f_0. Here an "intermediate structure" refers to a truss geometry before all the struts have been assembled. For the geometric and material properties, representative values are chosen for carbon fiber-reinforced plastic (CFRP) struts with length $L = 1$ m, density $\rho = 1600$ kg/m^3, circular cross section with diameter $d = 5$ cm, and elastic properties $E_Y = 325$ GPa and and $\nu = 0.3$. Linear beam elements (B31) are used to mesh the geometry with 10 elements along each strut, and all six degrees of freedom are kinematically coupled between the connected struts at each node. An eigenvalue analysis is performed to compute the fundamental free-free natural frequency.

Algorithm 1: Radially Expanding Euler Paths

Input: Eulerian graph $G(V, E)$, node coordinates *pos*
Output: Euler path *epath*

```
epath ← [random node in G]
while epath not complete do
    u = epath(end)
    centroid = mean(pos(epath))
    neighbors = nodal neighbors of u sorted by Euclidean
      distance to centroid
    k = 1
    while node not added to epath do
        v = neighbors(k)
        if edge (u,v) is not a bridge then
            append v to epath
        else
            k = k + 1
        end
    end
end
return epath
```

3. Results

Fig. 3 plots the computed fundamental natural frequencies of intermediate structures from 25 sampled radially expanding and randomized Euler paths for both the 2D equilateral triangle and 3D tetrahedral truss. The natural frequencies are plotted against an assembly percentage that represents the ratio of struts traversed by the path to the total number of struts. Here the Hierholzer algorithm was used to compute randomized paths while Algorithm 1 was used to generate radially expanding paths from a randomly selected starting node. Table 1 highlights a comparison of the minimum natural frequency and compute time between the two sampling approaches.

The results of Fig. 3 shows that radially expanding paths dramatically increase the minimum natural frequency of intermediate structures, by up to ~10x and 100x for the equilateral triangle and tetrahedral truss, respectively. The underlying reason is that the radially expanding paths result in intermediate structures with minimal unsupported struts. By contrast, the random paths result in large regions of unsupported struts and hence lower natural frequencies, as illustrated by the comparison of vibration mode shapes in Fig. 4.

Additionally, Fig. 3 shows that radially expanding paths result in decreased variance in natural frequency than the randomized paths. This is attributed to the radial symmetry of the truss geometries of Fig. 1, as all the radially expanding paths have

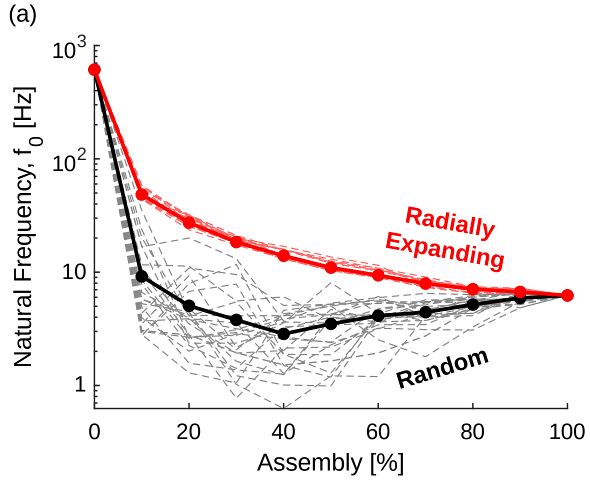

Fig. 3. Fundamental free-free natural frequency of intermediate structures from 25 radially expanding and randomized Euler paths for (a) 2D equilateral triangle and (b) 3D tetrahedral truss. Dashed lines plot data for all sampled paths and solid lines plot the average. Radially expanding paths result in structures with significantly higher natural frequency.

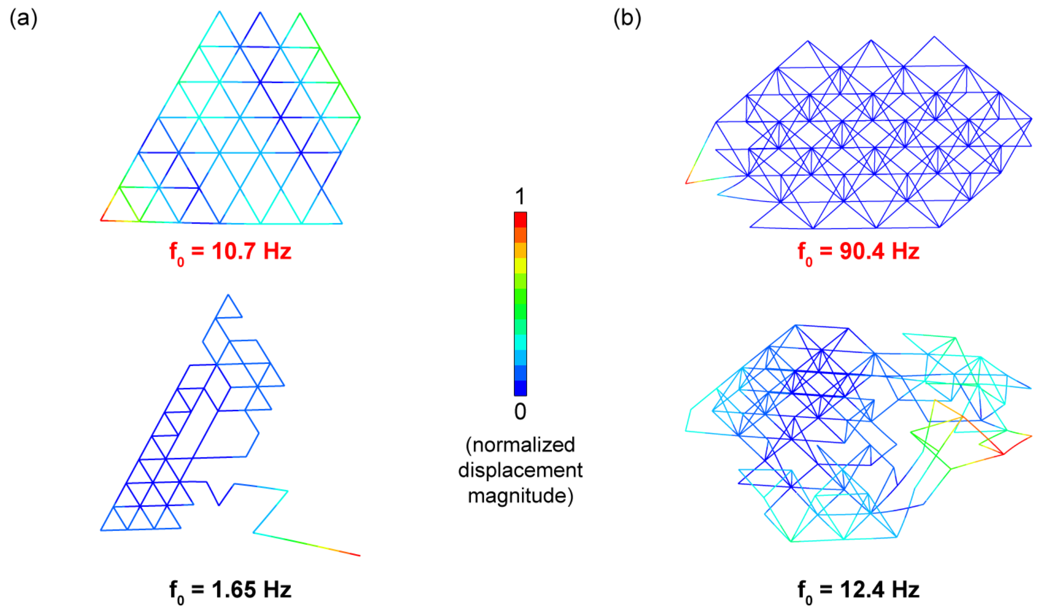

Fig. 4. Fundamental free-free vibration mode shapes from a radially expanding and randomized Euler path for (a) 2D equilateral triangle and (b) 3D tetrahedral truss at 50% assembly. The radially expanding path results in minimal unsupported struts.

Euler Path	Geometry	Minimum Natural Frequency, f_0 [Hz]	Compute Time [min]
Radially Expanding	Fig. 1a	6.23	110
	Fig. 1b	82.3	130
Random	Fig. 1a	0.63	100
	Fig. 1b	0.84	110

Table 1. Comparison of minimum natural frequency and compute time for the sampled radially expanding and random Euler paths in Fig. 3.

similar intermediate structures regardless of the starting node. Further, for the tetrahedral truss, radially expanding paths maintain a nearly constant natural frequency throughout the assembly sequence. This is because the vibration mode of each intermediate structure is dominated by the local deformation of a few unsupported struts around the circumference, unlike the global deformation of many unsupported struts for the randomized paths (cf. Fig. 4b). Together, the results of Fig. 3–4 demonstrate that radially expanding paths significantly improve intermediate stiffness and highlight the effectiveness of Algorithm 1 in sampling Euler paths for complex truss geometries.

4. Conclusion

This paper has presented an algorithm for generating radially expanding Euler paths through an undirected graph. The key differentiator of this algorithm is a locally optimal criterion for selecting nodes that minimize distance to the partially assembled geometry at each step. For the application of truss assembly, such paths are shown to dramatically increase the natural frequency of intermediate structures by up to two orders of magnitude as compared to randomized paths. Radially expanding Euler paths are therefore desirable for a wide range of assembly applications where intermediate stiffness is required, such as ground-based robotic construction and in-space assembly and manufacturing.

One caveat of the algorithm presented in this paper is that it is not as efficient as the classical Hierholzer algorithm for Euler paths and runs with complexity $O(n^2)$ where n is the number of edges. Future work incorporating the selection criterion into the Hierholzer algorithm may decrease the complexity to $O(n)$. Additionally, this work motivates research into other sampling strategies for truss assembly sequences to further explore the design space and develop heuristics for optimizing multiple performance metrics, e.g., stiffness, surface precision, and collision avoidance.

5. Acknolwedgements

H.G.B gratefully acknowledges financial support from the NASA Space Technology Graduate Research Opportunities (NSTGRO) program, through Grant No. 80NSSC23K1195.

References

1) N. Melenbrink, J. Werfel, and A. Menges, "On-site autonomous construction robots: Towards unsupervised building," *Automation in construction*, vol. 119, p. 103312, 2020.
2) H. G. Bhundiya, F. Royer, and Z. Cordero, "Engineering framework for assessing materials and processes for in-space manufacturing," *Journal of Materials Engineering and Performance*, vol. 31, no. 8, pp. 6045–6059, 2022.
3) B. Jenett, A. Abdel-Rahman, K. Cheung, and N. Gershenfeld, "Material–robot system for assembly of discrete cellular structures," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4019–4026, 2019.
4) W. Doggett, J. Dorsey, T. Jones, M. Mikulas, J. Teter, and D. Paddock, "Tritruss: a new and novel structural concept enabling modular space telescopes and space platforms," in *International Astronautical Congress*, no. NF1676L-32522, 2019.
5) Z. Xu, C. McCann, and A. M. Dollar, "Reconfigurable modular chain: A reversible material for folding three-dimensional lattice structures," *Journal of Mechanisms and Robotics*, vol. 9, no. 2, p. 025002, 2017.
6) H. G. Bhundiya and Z. C. Cordero, "Bend-forming: A cnc deformation process for fabricating 3d wireframe structures," *Additive Manufacturing Letters*, vol. 6, p. 100146, 2023.
7) D. Jungnickel and D. Jungnickel, *Graphs, Networks and Algorithms*, vol. 3. Springer, 2005.
8) M. S. Lake, L. D. Peterson, and M. B. Levine, "Rationale for defining structural requirements for large space telescopes," *Journal of Spacecraft and Rockets*, vol. 39, no. 5, pp. 674–681, 2002.
9) L. S. Homem-de Mello, "Sequence planning for robotic assembly of tetrahedral truss structures," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 25, no. 2, pp. 304–312, 2002.
10) D. C. Loudermilk, J. R. Cooper, and M. W. Otte, "Assembly sequence optimization for space structures," in *AIAA Aviation Forum and ASCEND 2024*, p. 4907, 2024.
11) Y. Wang and J. Liu, "Chaotic particle swarm optimization for assembly sequence planning," *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 2, pp. 212–222, 2010.
12) A. Mishra and S. Deb, "Assembly sequence optimization using a flower pollination algorithm-based approach," *Journal of Intelligent Manufacturing*, vol. 30, no. 2, pp. 461–482, 2019.
13) W. Cook and A. Rohe, "Computing minimum-weight perfect matchings," *Informs Journal on Computing*, vol. 11, no. 2, pp. 138–148, 1999.
14) C. Hierholzer and C. Wiener, "Über die möglichkeit, einen linienzug ohne wiederholung und ohne unterbrechung zu umfahren," *Mathematische Annalen*, vol. 6, no. 1, pp. 30–32, 1873.