**Abstract**

We wish to decide whether a simply connected flexible polygonal structure can be realized in Euclidean space. Two models are considered: polygonal linkages (body-and-joint framework) and contact graphs of unit disks in the plane. (1) We show that it is strongly NP-hard to decide whether a given polygonal linkage is realizable in the plane when the bodies are convex polygons and their contact graph is a tree; the problem is weakly NP-hard already for a chain of rectangles, but efficiently decidable for a chain of triangles hinged at distinct vertices. (2) We also show that it is strongly NP-hard to decide whether a given tree is the contact graph of interior-disjoint unit disks in the plane.

# 1 Introduction

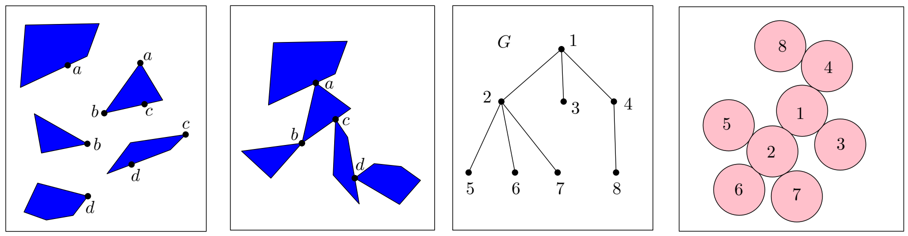
In this paper, we study the realizability of complex structures that are specified by their local geometry. The complex structures are represented as graphs with constraints on the separation between their vertices, and we ask if these graphs can be embedded in the plane subject to the constraints. We consider two models in the plane; refer to Fig. 1.

1. A **polygonal linkage** is a set $\mathcal{P}$ of convex polygons, and a set $H$ of hinges, where each hinge $h \in H$ corresponds to two or more points on the boundaries of distinct polygons in $\mathcal{P}$. A **realization** of a polygonal linkage is an interior-disjoint placement of congruent copies of the polygons in $\mathcal{P}$ such that the points corresponding to each hinge are identified. A **realization with orientation** uses only translated or rotated copies of the polygons in $\mathcal{P}$ (no reflections) and for each hinge, the cyclic order of incident polygons is given. The topology of a polygonal linkage can be represented by the **hinge graph**, a bipartite graph where the vertices correspond to polygons in $\mathcal{P}$ and the hinges in $H$, and edges represent the polygon-hinge incidences.

2. An (abstract) graph is a **coin graph** if it is the intersection graph of a set of interior-disjoint unit disks in the plane (where the vertices correspond to disks and two vertices are adjacent if and only if the corresponding disks are in contact). A **coin graph with embedding** is a coin graph *together with* a cyclic order of the neighbors for each vertex (i.e., each disk).

The POLYGONAL LINKAGE REALIZABILITY (PLR) problem asks whether a given polygonal linkage admits a realization; and PLR WITH FIXED ORIENTATION asks whether it admits a realization with a given orientation. The COIN GRAPH RECOGNITION (CGR) problem asks whether a given (abstract) graph $G$ is the contact graph of interior-disjoint unit disks in the plane; and CGR WITH FIXED EMBEDDING asks whether a given plane graph $G$ is the contact graph of interior-disjoint unit disks in the plane with the same counterclockwise order of neighbors at each vertex.

These problems, in general, are known to be NP-hard (see details below). However, the hardness reductions crucially rely on graphs with a large number of cycles. We revisit these problems for simply connected topologies, where the hinge graph and the coin graph are trees.




(a) (b) (c) (d)

Figure 1: (a) A set of convex polygons and hinges. (b) A realization of the polygonal linkage (with fixed orientation). (c) A graph $G$ with 8 vertices. (d) An arrangement of interior-disjoint unit disks whose contact graph is $G$.

**Summary of results.** Our main result is that the realizability problem remains NP-hard for simply connected polygonal linkages, the only exceptions are chains of triangles or rectangles hinged at distinct vertices. In an attempt to identify the most general problem that is not NP-hard, we considered several variants. Some variants are always realizable, some have easy hardness reductions, and some reductions required substantial new machinery. Our most demanding result is the NP-hardness of the recognition of *coin trees* with fixed embedding. We summarize the results here.

1. We start with *chains of polygons*, that is, polygonal linkages in which the hinge graph is a path (Section **??**). It is easy to see that every chain of triangles or rectangles hinged together at distinct vertices is realizable and a realization can be computed efficiently. However, the problem becomes weakly NP-hard for chains of convex quadrilaterals hinged at distinct vertices or for chains of triangles where one hinge may be at anywhere on the boundary. Our reduction uses PARTITION.

2. We show that PLR (with arbitrary orientation) is strongly NP-hard when the hinge graph is a tree, using an easy reduction from 3SAT with the classic logic engine method (the easy proof is available in the full paper). The reduction crucially depends on possible reflections of the polygons.

3. We show that PLR with fixed orientation is also strongly NP-hard when the hinge graph is a tree (Section **??**), using a significantly more involved reduction from PLANAR3SAT. We carefully design gadgets for variables, clauses and a planar graph to simulate PLANAR3SAT.

4. We reduce the recognition of coin trees with fixed embedding to the previous problem (PLR with fixed orientation), by simulating suitable polygons with an arrangement of unit disks (Section **??**). It would be easy to model a polygon by a *rigid* coin graph (e.g., a section of the triangular grid), but all rigid graphs induce cycles. The main technical difficulty is that when the coin graph is a tree, any realization with unit disks is highly flexible, and simulating a rigid object becomes a challenge. We construct coin trees with "stable" realizations, which may be of independent interest.

**Related Previous Work.** Previous research has established NP-hardness in several easy cases, but realizability for simply connected structures remained open. Polygonal linkages (or body-and-joint frameworks) are a generalization of classical linkages (bar-and-joint frameworks) in rigidity theory. A linkage is a graph $G = (V, E)$ with given edge lengths. A realization of a linkage is a (crossing-free) straight-line embedding of $G$ in the plane. Based on ideas developed by Bhatt and Cosmadakis [**?**], who proved that the realizability of linkages is NP-complete on the integer grid, the *logic engine* method [**?**, **?**, **?**, **?**] has become a standard tool for proving NP-hardness in graph drawing. The logic engine is a graph composed of rigid 2-connected components, where two possible realizations of a 2-connected component encode a binary variable.

However, the logic engine method is **not** applicable to problems with fixed embedding or orientation, where the circular order of the neighbors of each vertex is part of the input. Cabello et al. [**?**, **?**] used a significantly more elaborate reduction to show that the realizability of 3-connected linkages (where the orientation is unique by Whitney's theorem [**?**]) is NP-hard. This problem is efficiently decidable, though, for near-triangulations [**?**, **?**].

Note that every *tree* linkage can be realized in $\mathbb{R}^2$ with almost collinear edges. According to the celebrated *Carpenter's Rule Theorem* [**?**, **?**], every realization of a path (or a cycle) linkage can be continuously moved (without self-intersection) to any other realization. In other words, the realization space of such a linkage is always connected. However, there are trees of maximum degree 3 with as few as 8 edges whose realization space is disconnected [**?**]; and deciding whether the realization space of a tree linkage is connected is PSPACE-complete [**?**]. (Earlier, Reif [**?**] showed that it is PSPACE-complete to decide whether a polygonal linkage can be moved from one realization to another among polygonal obstacles in $\mathbb{R}^3$.) Cheong et al. [**?**] consider the "inverse" problems of introducing the minimum number of point obstacles to reduce the configuration space of a polygonal linkage to a unique realization.

Connelly et al. [**?**] showed that the Carpenter's Rule Theorem generalizes to certain polygonal linkages obtained by replacing the edges of a path linkage with special polygons (called *slender adornments*). Our Theorem **??** indicates that if we are allowed to replace the edges of a linkage with arbitrary convex polygons, then deciding whether the realization space is empty or not is already NP-hard.

Recognition problems for intersection graphs of various geometric object have a rich history [**?**]. Breu and Kirkpatrick [**?**] proved that it is NP-hard to decide whether a graph $G$ is the contact graph of unit disks in the plane, i.e., recognizing *coin graphs* is NP-hard; see also [**?**]. Recognizing outerplanar coin graphs is already NP-hard, but decidable in linear time for caterpillars [**?**]. It is also NP-hard to recognize the contact graphs of pseudo-disks [**?**] and disks of bounded radii [**?**] in the plane, and unit disks in higher dimensions [**?**, **?**]. All these hardness reductions produce graphs with a large number of cycles, and do not apply to trees. Note that the contact graphs of disks *of arbitrary radii* are exactly the planar graphs (by Koebe's circle packing theorem), and planarity testing is polynomial. Consequently, every tree is the contact graph of disks of *some* radii in the plane.

Eades and Wormald [**?**] showed that it is NP-hard to decide whether a given tree is a *subgraph* of a coin graph. Schaefer [**?**] proved deciding whether a graph with given edge lengths can be realized by a straight-line drawing (possibly with crossing edges)