Complex structures in nature are often composed of elementary pieces that obey simple local composition rules. Molecular biology, nanomanufacturing, and self-assembly are prime examples. Mathematical models for this phenomenon typically rely on rigidity theory and formal languages. In this paper, we study the realizability of complex structures that are given with a local specification. We consider two models in Euclidean plane.

1. A **polygonal linkage** is a set $\mathscr{P}$ of convex polygons, and a set H of hinges, where each hinge $h \in H$ corresponds to two points on the boundary of two distinct polygons. A *realization* of a polygonal linkage is an interior-disjoint placement of congruent copies of the polygons in $\mathscr{P}$ such that the points corresponding to each hinge are identified (Fig. 1, left).

2. A **disk arrangement** is a set $\mathscr{D}$ of pairwise interior-disjoint disks in the plane. The contact graph of a disk arrangement $\mathscr{D}$ is a graph $G = (\mathscr{D}, E)$ where two vertices are adjacent if the corresponding disks intersect (kiss). A *realization* of a vertex-weighted graph G as a contact graph of disks is a disk arrangement whose contact graph is G and the radius of each disk is the corresponding vertex weight.

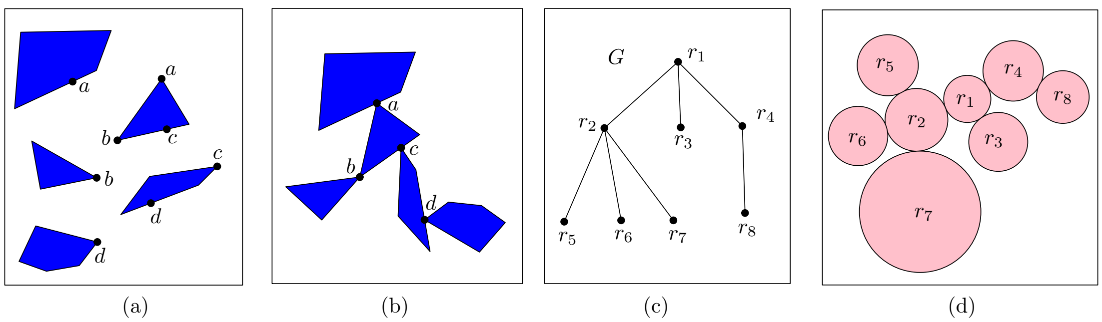


Figure 1: (a) A set of convex polygons and hinges. (b) A realization of the polygonal linkage from (a). (c) A graph G with vertex weights $r_1, \ldots, r_8$. (d) A disk arrangement that realizes the weighted graph G as a contact graph with radii equal to the corresponding weights.

Each model has two variants, depending on whether *reflection* is allowed for the realization of each piece independently. For polygonal linkages, an *oriented realization* requires translated and rotated copies of the polygons in $\mathscr{P}$ (i.e., reflection is not allowed). An *ordered contact graph* for a disk arrangement is a *plane graph* G, where the circular order of the neighbors of each vertex is specified, and an *oriented realization* is disk arrangement with the given ordered contact graph.

The *realizability* problem for a polygonal linkage asks whether a given polygonal linkage has a realization (resp., orientated realization). For a weighted planar (resp., plane) graph,, it asks whether the graph is the contact graph (resp., ordered contact graph) of some disk arrangement with specified radii. These problems, in general, are known to be NP-hard. Specifically, it is NP-hard to decide whether a given planar (or plane) graph can be embedded in $\mathbb{R}^2$ with given edge lengths [**?**, **?**]. Since an edge of given length can be modeled by a suitably long and skinny rhombus, the realizability of polygonal linkages is also NP-hard. The recognition of the contact graphs of unit disks in the plane (a.k.a. coin graphs) is NP-hard [**?**], and so the realizability of weighted graphs as contact graphs of disks is also NP-hard. However, previous reductions crucially rely on configurations with high genus: the planar graphs in [**?**, **?**] and the coin graphs in [**?**] have many cycles.

In this paper, we consider the above four realizability problems when the union of the polygons (resp., disks) in the desired configuration is simply connected (i.e., contractible). That is, the contact graph of the disks is a tree, or the "hinge graph" of the polygonal linkage is a tree (the vertices in the *hinge graph* are the polygons in $\mathscr{P}$, and edges represent a hinge between two polygons). Our main result is that realizability remains NP-hard when restricted to simply connected structures.

Theorem 0.1. *It is NP-complete to decide whether a polygonal linkage whose hinge graph is a tree can be realized (both with and without orientation).*

Theorem 0.2. *It is NP-complete to decide whether a given tree (resp., plane tree) with positive vertex weights is the contact graph (resp., ordered contact graph) of a disk arrangements with specified radii.*

The unoriented versions, where the underlying graph (hinge graph or contact graph) is a tree can easily be handled with the logic engine method (Section **??**). We prove Theorem 0.1 for *oriented* realizations with a reduction from PLANAR-3SAT (Section **??**), and then reduce the realizability of ordered contact trees to the oriented realization of polygonal linkages by simulating polygons with arrangements of disks (Section **??**).

Related Previous Work. 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. Bhatt and Cosmadakis [**?**] proved that the realizability of linkages is NP-hard. Their "logic engine" method [**?**, **?**, **?**, **?**], has become a powerful tool in graph drawing. The logic engine is a graph composed of rigid 2-connected components, connected by cut vertices (hinges). The two possible realizations of each 2-connected component (that differ by a single reflection) represent the truth assignment of a binary variable. This method does not applicable to the *oriented* version of the realizability, where the circular order of the neighbors of each vertex is part of the input. Cabello et al. [**?**, **?**] proved that the realizability of 3-connected linkages (where the orientation is unique by Steinitz's theorem) is NP-hard, but efficiently decidable 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 at 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. [**?**] considers 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, which are obtained by replacing the edges of a path linkage with special polygons called (*slender adornments*). Our Theorem 0.1 indicates that if we are allowed to replace the edges of a path 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 (a.k.a. recognizing *coin graphs* is NP-hard). A simpler proof was later provided via the logic engine [**?**]. 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 of high genus, and do not apply to trees. Note that the contact graphs of disks (of arbitrary radii) are exactly the planar graph (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.

1 Linkages and Polygonal Linkages

Given a *graph*, an ordered pair $G = (V,E)$, comprising of a set V of vertices or nodes together with a set E of edges or lines, then a *linkage* of G is the realization (or embedding) of G in $\mathbb{R}^2$. A *polygonal linkage* is an ordered pair, $L = (H,P)$, comprises of a set of polygons, P, and a set of hinge points H where each hinge $h \in H$ corresponds to two points on the boundary of two distinct polygons in P. Without loss of generality, for this paper, we focus on linkages and polygonal linkages that are simple planar graphs, i.e.:

(i) does not have edges (polygons) that cross (intersect),

(ii) have loops (i.e. $(v,v) \in E$), or

(iii) does not have multiple edges between any pair of vertices.

We may visit special cases in which we look at planar graphs that satisfy the last two conditions but not the first.

1.1 Linkage Realization

1.2 Configuration Spaces of Linkages

To describe the types of motion that we are interested in linkages we must define the graph isomorphism. Two graphs $G = (V_1, E_1)$ and $\Gamma = (V_2, E_2)$, a bijection $f : V_1 \mapsto V_2$ such that for any two vertices $u, v \in V_1$ that are adjacent, i.e. $(u, v) \in E_1$, if and only if $(f(u), f(v)) \in E_2$.

Graph	Vertices	Edges
G	$\{a,b,c,d,e\}$	$\{(a,b),(b,c),(c,d),(d,e),(e,a)\}$
Γ	$\{1,2,3,4,5\}$	$\{(1,2),(2,3),(3,4),(4,5),(5,1)\}$

Table 1: Two graphs that are isomorphic with the alphabetical isomorphism $f(a) = 1$, $f(b) = 2$, $f(c) = 3$, $f(d) = 4$, $f(e) = 5$.

Next we add restrictions to our graph isomorphisms to narrow our focus:

(i) We focus on isomorphisms for planar graphs and or polygonal linkages, simple planar graphs, and

(ii) the isomorphism preserves edge lengths (polygonal area), e.g. $d(u, v) = d(f(u), f(v))$.

With these restrictions of our isomorphisms, we can begin to describe a range of motion to transform a linkage. That range of motion is said to be the configuration space of that linkage. To expand on this concept, for given linkage, $L = (V, E)$, and for a given vertex $v \in V$, the set of points in which v can be realized in the plane would be the configuration space for that vertex, C_v. Defining some order of the vertices in L, i.e. $V = \{v_n\}_{i=1}^{n}$, then the *configuration space* for L is said to be the cartesion product of the configuration space of vertices:

$$C(L) = C_{v_1} \times C_{v_2} \times \cdots \times C_{v_n} \tag{1}$$

Some food for thought on configuration spaces and motions on linkages:

(i) A configuration space is said to be *connected* if there is a continuous mapping for any two planar realizations (linkages) of a graph in the plane. Otherwise it is said to be *disconnected*.

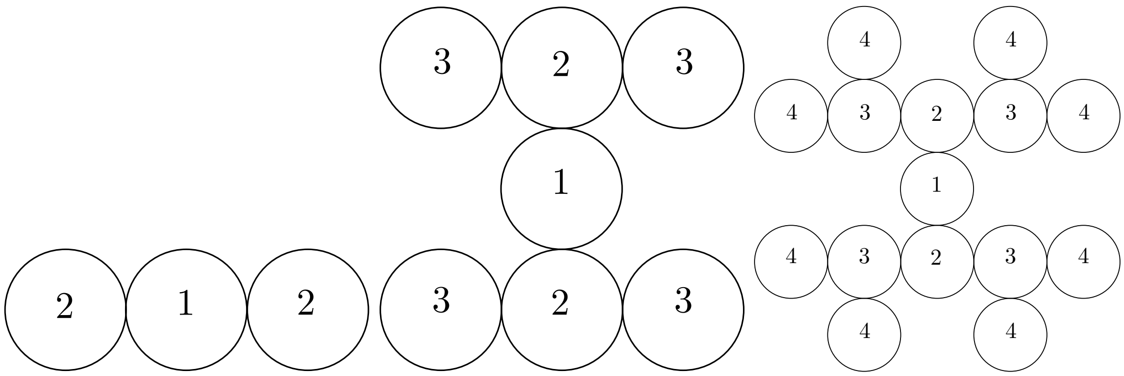
(ii) If the configuration space of a vertex, C_v, is a singleton set, then the vertex is said to be *pinned*. Otherwise it is said to be *free*.

(iii) The types of motions (mappings) that we refrain from using on linkages are translations.

2 Disk Arrangements

2.1 Oriented Realizations

2.2 Disk Packing Confinement Problem



(a) A disk arrangement with two layers of disks (b) A disk arrangement with three layers of disks (c) A disk arrangement with four layers of disks

Figure 2: The gradual growth of disk arrangements by adding two kissing disks to each of the previously generated disks. By continuing this arrangement growth, the space needed to contain the kissing disks will exceed the area containing the disk arrangements.

Here goes something

2.3 Circle Packing

It turns out the circle packings are an equivalent way to to represent linkages and their corresponding problems. Before we establish the relation, we will cover some fundamental concepts of circle packings. A *circle packing*, P, embedded in a plane is a set of circles with disjoint interiors $\{C_i\}_{i=1}^n$ such that for any circle $C \in \{C_i\}_{i=1}^n$, C is tangent to a different circle of $\{C_i\}_{i=1}^n$.

Any circle embedded in a plane has a given center point and radius. This information of planar embedded circle packings allows us to establish the relationship to linkages with the following construction:

(i) let the centerpoints of the circle packing be a set of vertices V;

(ii) if two circles in a circle packing are tangent, we define an edge between their centerpoints. The distance of this edge is the sum the radii of the two tangent circles.

This construction establishes a relationship between linkages and circle packings. It begs questioning as to whether every connected simple planar graph has a circle packing. The question is answered in the following theorem.

Theorem 2.1 (Circle Packing Theorem)**.** *For every connected simple planar graph G there is a circle packing in the plane whose intersection graph is (isomorphic to) G.*

A proof of Theorem 2.1 is found in chapter 7 of [**?**]. Theorem 2.1 also gives us the ability to establish an equivalent definition of configuration spaces on circle packings and allows us to pose the same realizability problems found with simple planar graphs. To narrow the focus of the types of circle packing realizabilty problems that we are interested in, we add the following restriction: all circles in a circle packing have unit diameter.

2.3.1 Realizability Problems in Unit Disk Packings

In [**?**], it was shown that unit disk graph recognition is NP-Hard.

2.4 Area Packing Problem

2.4.1 Hinged Polygons

Definition 2.1 (Polygonal Chain)**.** A polygonal chain $P = (v_0, v_1, \ldots, v_{n-1})$ is a sequence of consecutively joined segments (or edges) $e_i = v_i v_{i+1}$ of fixed lengths $l_i = |e_i|$, in a plane. [**?**]

A chain is said to be closed if $v_{n-1} = v_1$, otherwise it is said to be open. Hinged polygons have been researched for decades and related to linkage problems [**?**, **?**].

Consider the locked configuration of figure 4. We can configure the hexagons to be locked by placing hinged points as follows:

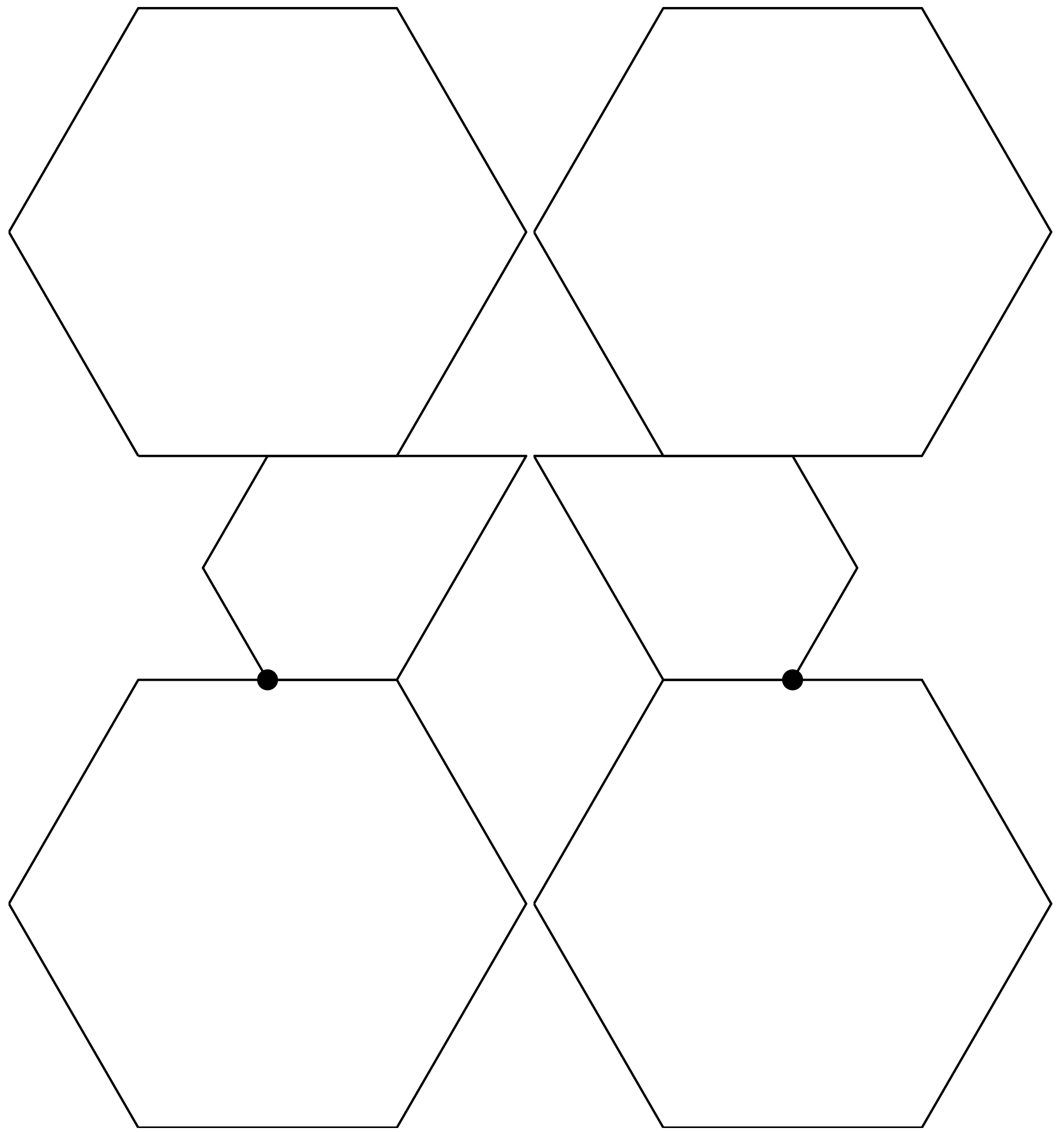
2.4.2 Hinged Hexagons of Fixed Size

The Shapes Figure 5 is a locking shape: Figure 5 shall reside in the boundary of a lattice and have a hinge point at one vertex where the locking shape and boundary meet.

Junctions We define junctions to be the point three hexagons meet in a hexagonal lattice, e.g. Figure 9.

Central Scaling

Junctions in Conjunctive Normal Form Explain the configurations we're interested in.



(a) blah blah blah (b) blah blah blah

Figure 3: Due to the strip in the plane that the hexagon is bounded within the configuration space is limited to just two realizations.

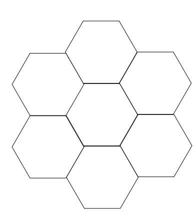


Figure 4: A locked 7 hexagonal configuration. (needs to modify picture by placing red points for hing points.)

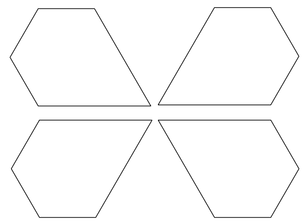


Figure 5: A locking shape in the lattice boundary's channel.

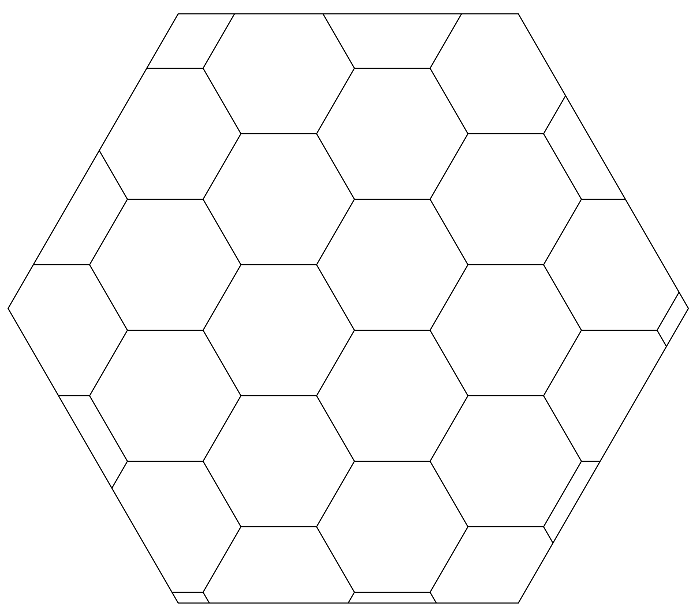
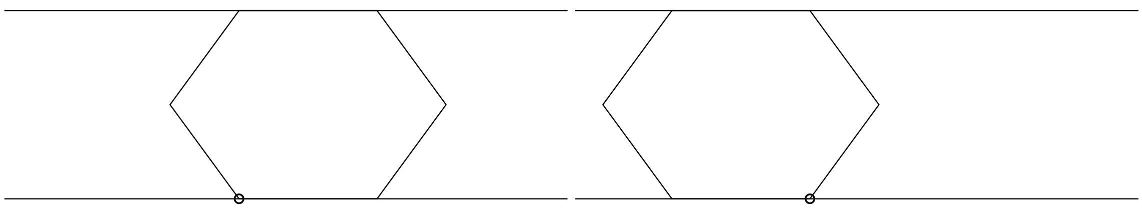


Figure 6: A portion of a hexagonal lattice.

3 Configuration Spaces

3.0.3 Confining Linkages to a Restricted Space Within a Configuration Space

So we've covered the idea of linkages within a plane; now let's constrain the plane to a strip and have a linkage that is a *polygon*, i.e. a linkage that forms a closed chain (e.g. Table **??**), hugging the boundaries of the strip: So here we have a linkage whose conifguration space is limited to just two realizations. With just

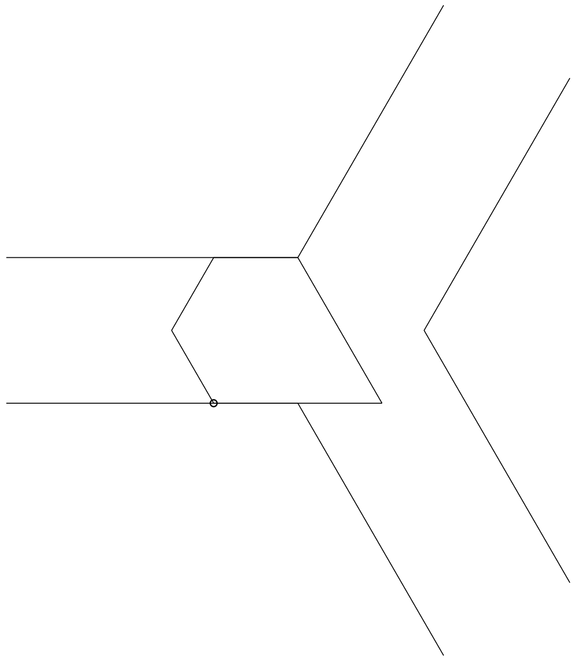


(a) A bounded hexagon that resides in a channel with a pinned vertex

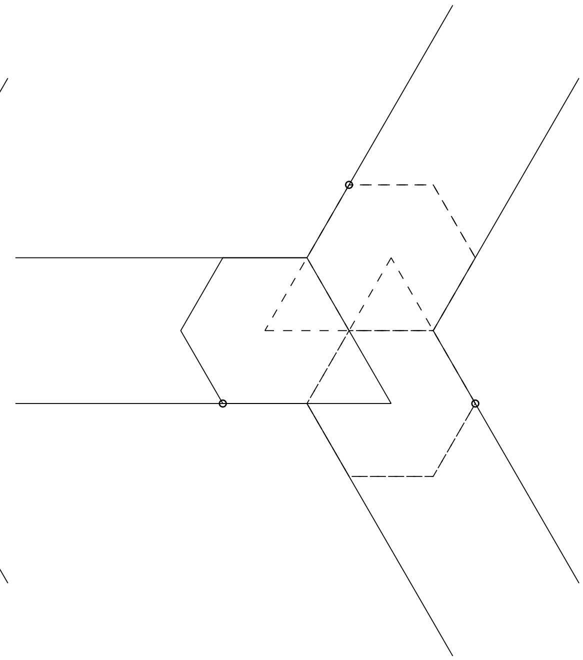
(b) The second realization of the hexagon residing in a channel with a pinned vertex.

Figure 7: Due to the strip in the plane that the hexagon is bounded within the configuration space is limited to just two realizations.

two realizations, we can assign a binary value to them and have the linkage act as a boolean variable. We will revisit this concept when we cover satisfiability problems later on in the paper.



(a) A pentagon that is pinned in a channel junction that is formed by the sides of 3 large regular hexagons. It has two possible configurations, much like that of 7



(b) A pinned pentagon residing in a channel junction that is formed by the sides of 3 large regular hexagons with 2 dashed pentagons intersecting it.

Figure 8: Suppose the channel formed is a junction of three regular hexagons. The polygon partially residing in the junction is a regular hexagon with an equalateral triangle appended at an edge. This polygon would prevent other polygons (i.e. the dashed polygons) of the same shape residing in the center of the channel without intersection. This demonstrates that a the configuration space within a multichannel environment can have concurrency issues, i.e. some configurations cannot be realizable.

Expanding upon the ideo of 7, forming channels with junctions as shown in Figure **??** can be formed as such by evenly spacing the edges of a hexagonal lattice. Visually, it is shown that only one of three possible pentagons can reside in the channel at one time. By asserting certain conditions on the lattice, and extending the problem to a greater region of a hexagonal lattice, we will be able to pose a realizability problem of whether a configuration $\mathscr{A}$ can be reconfigured to $\mathscr{B}$ by switching pentagons without violating overlapped polygon conditions.

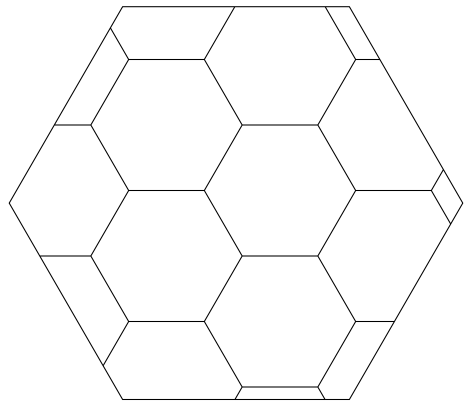


Figure 9: A hexagonal lattice contained in a hexagon.

3.0.4 Realizability of Linkages

Suppose we had two configurations of a linkage, $\mathscr{A}$ and $\mathscr{B}$. A question that can be posed is can we reconfigure $\mathscr{A}$ to $\mathscr{B}$ continuously while respecting simple planar graph conditions? The answer to this question is a yes or no. It has been shown that this problem can be posed as a planar satisfiability problem [**?**, **?**] (Later on in this paper we'll cover satisfiability problems). This is the type of problem that we face in this paper. We will continue to explore this in a different manner, with circle packings.