GENERALIZED CIRCULAR MEDIAL GRAPHS

NICK REICHERT

Abstract. This paper gives a discussion of medial graphs in the non-planar case, as described by [4]. It also proposes some ideas for embedding extension problems and the genus problem in topological graph theory. In addition, it provides a (very slow) algorithm for determining the genus of a graph.

Contents

1. Introduction

We have very good results for the recoverability of circular planar graphs. Is it possible to generalize those results? One way to attack the general recoverability problem is by examining medial graphs in the non-planar case.

2. Representing Non-Planar Graphs

2.1. Basic Representation. In the circular planar case, visual representations of medial graphs were extremely useful. For example, it is very clear from a drawing

Date: August 17, 2004.

whether a medial graph has a lens, and thus whether or not it is recoverable. Thus, if we are to try generalize those results in a visual sense, we must define what we mean by the medial graph in the non-planar case. First, though, we describe a way to represent non-planar graphs so that drawing medial graphs will be somewhat easier.

Definition 2.1. The *genus* of a graph G is the smallest number n such that G can be drawn on a surface S of genus n without any edge crossings. We consider, for the most part, only the case that S is orientable, though the non-orientable case is very interesting (see [1]).

Now, we must decide on a convention to represent non-planar graphs. Attempting to draw a graph on a "flattened" planar representation of an arbitrary surface is troublesome. Though it is useful to be able to draw graphs in such a way in certain, limited cases, it is often very time consuming and difficult to understand once completed. This problem is heightened as the genus of the surface increases and as the graphs increase in complexity. Using dashed lines to represent edges on hidden surfaces on the torus works within reason, but beyond that case this method is, by and large, extremely impractical.

[5] outlines an alternative way to represent non-planar graphs with is much simpler and less confusing. In an attempt to make this paper somewhat self-contained, his argument is reproduced here with some additional discussion.

One can make two cuts on a torus to reduce it to a rectangle—one equatorial, and another around and through the loop. If we identify opposite sides of the rectangle as shown in figure 1, then we get a very simple way to represent the torus.

It is critically important to indentify opposite edges in a way consistent with figure 1; specifically, if we place cartesian coordiantes on the rectangle, points with equal vertical coordinate on the vertical edges are identified, and points with equal horizontal components on the horizontal edges are identified. In other words, the tip of an arrow on one side is identified with the tip of an arrow on the opposite one, with the likewise convention for the tails of arrows. If this convention is not followed, non-orientable surfaces, the klein bottle and the projective plane, arise. Figure 2 gives an example of the complete graph on five nodes drawn on the rectangular representation of the torus so that edges are identified in the correct way.

Now that we have a good way to represent the torus, we can extend this method to all orientable surfaces of finite genus. Make a circular hole in each of three tori, so that the circle that is cut out is simply connected. Then no matter how we move the circles around on each torus (provided we don't move "over a hole") we can still represent each of these new surfaces on the rectangle with edges identified. So if we move the circles as in figure 3, then we can represent these special tori as hexagons by opening the circle, so to speak. Then, connect the three hexagons at their sides which do not have identified points to obtain the dodecagon shown at the bottom of figure three. Then, since what we just did is equivalent to joining three tori, we have a way to represent a surface of genus three.

This method easily generalizes to other genera. A surface of genus g is representable as a $4n$-gon, partitioned into sets of four adjacent edges so identified in a way analagous to the torus. We will call this the *planar representation* of the

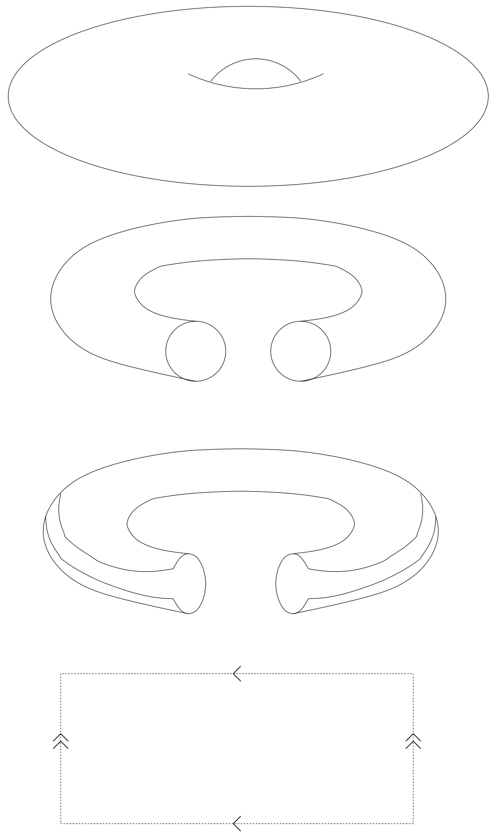

FIGURE 1. A Sequence of Cuts From A Torus to A Rectangle, With Edges Identified

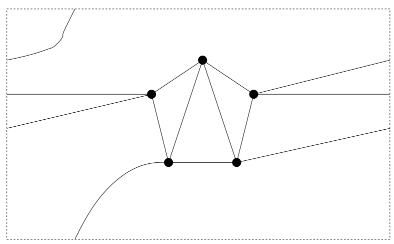

FIGURE 2. A Genus-1 Drawing of the Complete Graph on Five Nodes.

surface.

One natural question to ask is: what can we say about a $4n+2$-gon?

Proposition 2.2. *Consider a $4n$-gon, with edges identified in the way we have defined. Suppose we insert a pair of edges between two of the sets of four identified edges. Identify the new pair of edges is identified as shown in figure 4. Then if a*

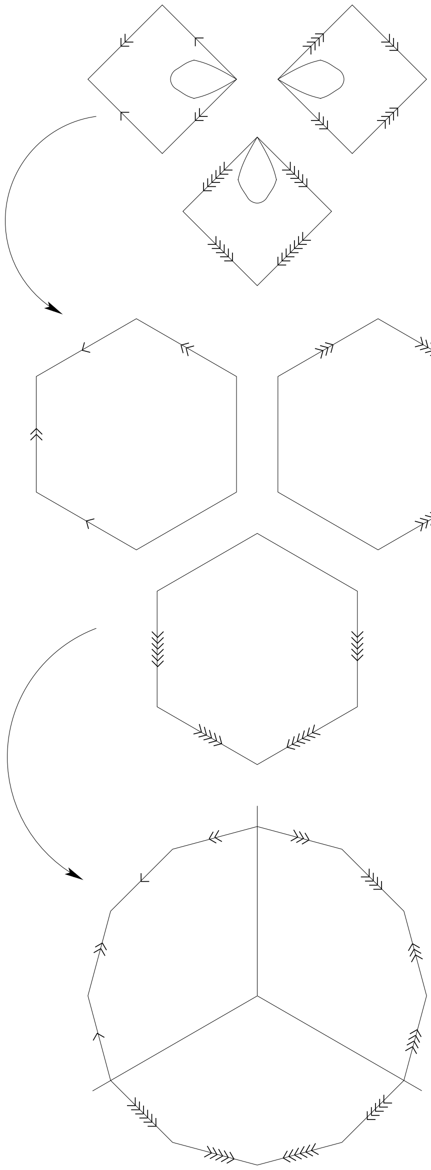

FIGURE 3. Joining Three Tori to Get a Genus Three Surface

graph can be drawn on that surface without edges crossing, then that graph can be drawn on the conventional $4n$*-gon.*

Proof. One way to see this is to note that a cylinder can be represented as a rectangle, with only one pair of opposite sides identified. Then, using the gluing process shown in figure 3, we see that the $4n + 2$-gon, as we have defined it, is a surface of genus n with a cylinder attached. But that surface topologically equivalent to a surface of genus n that has been punctured twice. However, we can make those puncture holes as small as we please, so they will not obstruct any embeddings

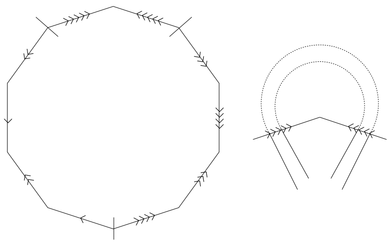

FIGURE 4. A $4n+2$-gon, and How to See Its Equivalence to a $4n$-gon

on the $4n$-gon, nor will they allow any additional graphs to be embedded in the surface.

An alternative, much simpler way to see this is to note that edges going into the extra pair of sides come out reversed—that is, we can accomplish the same result by simply "turning the edges around" (see figure 4). □

If we identify the new pair of edges in our $4n+2$-gon in a different way than as defined in Proposition 2.2, that is, we make a "twist", we get a very interesting surface: a mobius strip adjoined to a surface of genus n. On such a surface, is it possible to go from the outside of the surface to the inside of the surface, or is that inconsistent with our conventions? If we can go from the outside to the inside, have we effectively doubled the number of holes we have to work with? In other words, given a graph with genus $2n$, can we embed it in this new type of $4n+2$-gon?

2.2. Determining The Genus And Embeddings of Non-Planar Circular Graph Graphs.

Definition 2.3. Let G be a graph with boundary. Suppose n is the smallest genus an surface can have so that G can be embedded on the surface, with the additional condition that the boundary of G lie on a circle whose interior on that surface is simply connected, and that no part of any edge of G lies within the circle. Then we say that G is *genus-n circular*. A graph H without boundary is said to be *genus-n circular* if the graph H' that is H with all nodes declared to be boundary nodes is genus-n circular.

Remark 2.4. When G is genus-0 circular, we shall simply say that it is *circular planar*. To justify this terminology, we must show that G is genus-0 circular iff we can embed G so that its boundary lies on a circle, and so that no edge of G lies outside the circle (thus a genus-0 circular graph would be circular planar in the traditional sense). To see this, simply reflect about the circle (e.g. declare the origin of the complex plane to be the center of the circle, and the radius of the circle to be one, then take the map $1/z$). Likewise, a circular planar graph (in the convential sense) is genus-0 circular. Figure 6 shows an example.

In the planar case, we saw that, visually, medial graphs were very dependent on embeddings. Here we make the term "embedding" precise.

Definition 2.5. Let $G = (V, E)$ be a simple graph with v vertices. Let a_i be an ordered set with norm n_i, where n_i is the valence of v_i. Furthermore, the elements of a_i are all j such that v_i is adjacent to v_j. Moreover, let each element of a_i occur only once in a_i (so that every j satisfying v_i is adjacent to v_j occurs in a_i). Then

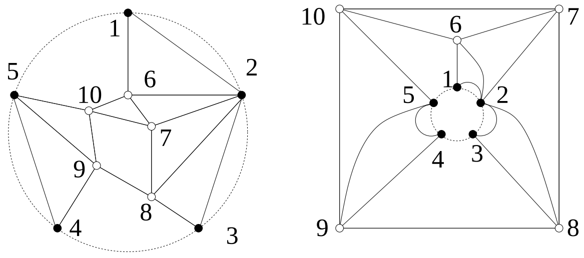

FIGURE 5. Two Different Embeddings of a Circular Planar Graph, One With All Edges Inside the Boundary Circle and the Other With All Edges Outside the Boundary Circle.

a *combinatorial embedding* C *of* G is $C = (G, A)$, where $A = \{a_1, a_2, ...a_v\}$. We will say that two combinatorial embeddings C_1 and C_2 of G are *equivalent* if each element of A_1 is a cyclic permutation of the corresponding element in A_2.

More colloquially, a combinatorial embedding is a graph with the ordering of edges around each node specified. This is why cyclic permutations of the a_i are allowed in equivalence—we could choose start with any edge around a node. For the remainder of this paper, we will assume that we always have a clockwise ordering of edges around each node in a combinatorial embedding.

Definition 2.6. Let $G = (V, E)$ be a simple graph with v vertices. Let x_i be such that if we draw G on the planar representation of a surface of genus n, then x_i is x-coordinate of v_i. Define y_i likewise. Then a *coordinate embedding* XY *of* G is $XY = (G, X, Y)$, where $X = \{x_1, x_2, ...x_v\}$ and $Y = \{y_1, y_2, ...y_v\}$.

Why do we define two types of embeddings? As it turns out, each type has its advantages.

Combinatorial embeddings are useful because they allow us to find very important properties of graphs, such as the genus, without having to actually draw the graph (see section 2.4). Furthermore, medial graphs are uniquely determined by combinatorial embeddings. Finally, there are a finite number of combinatorial embeddings for a finite graph (a fact also used in section 2.4). In particular, let (v_i) represent the degree of vertex i in a graph G. Then if G has v vertices, the total number of combinatorial embeddings γ is

$$\gamma = \prod_{1}^{v} (d(v_i) - 1)! \tag{1}$$

This comes from the fact that, for each node, there are $d(v_i)!$ ways to order the edges around the node. However, since we do not wish to count equivalent combinatorial embeddings more than once, we divide by the number of cyclic permutations of edges around each node. Since there are $d(v_i)!$ cyclic permutations per node, we arrive at the above formula.

Coordinate embeddings are not as elegant, but they are nonetheless important. If we are actually going to draw the medial graphs (or just graphs), we are going to need a coordinate embedding. A combinatorial embedding certainly isn't enough to determine a coordinate embedding; a difficult problem is to find a coordinate embedding from a combinatorial embedding. [6] gives a method for giving a coordinate embedding given a certain type of circular planar graph. Other than that, I have found very little information about finding coordinate embeddings of graphs.

Now that we have some definitions, we can state a result about genus-n circular graphs.

Proposition 2.7. *Let G be a graph with boundary, and with v nodes. Suppose G is genus-n circular. Then, in $O(v^3)$ time, we can determine both n and find a combinatorial embedding of G in a surface of genus n.*

To see this, we first note that finding n reduces to the topic in normal graph theory of finding the genus of a graph. To see this, refer to Lemma 1.4 of [2]. With slight modifications, we can apply it to the genus-n circular case.

Lemma 2.8. *Suppose $G(V, V_B, E)$ is a graph with boundary, and $V_B = \{V_1, V_2, ...V_n\}$. Let $H(V', E')$ be a graph (not a graph with boundary) so that $V' = V \cup P$, $E' = E \cup \{(P, V_1), (P, V_2), ...(P, V_n)\}$. Then G is genus-n circular iff H is genus n.*

Proof. First we prove necessity. If G is genus-n circular, then embed G in a surface of genus-n with its boundary lying on an empty, simply-connected circle C. Place P inside C. Then we have a genus-n embedding of H.

Second, we show sufficiency. If H is of genus n, move the nodes adjacent to P so that they lie on a small circle about H. We requrie that the interior of the circle be simply connected, and that the only edges inside the circle are those with one end at P. Then call the nodes adjacent to P boundary nodes. Delete P and all edges adjacent to it. Then we now have the graph G, with its boundary on an empty, simply connected circle. □

[1] describes a linear time algorithm for determining if a graph is embeddable in a surface of genus n. If so, it gives a combinatorial embedding for that genus. Now, the maximum genus of a graph with v boundary vertices is

$$\left\lceil \frac{(v-3)(v-4)}{12} \right\rceil \tag{2}$$

because that is the genus of K_v [3]. This is an $O(v^2)$ bound. So, since we have a linear time algorithm for checking and finding an embedding of a graph in an arbitrary surface, we have an $O(v^3)$ algorithm that satisfies the requirements of Proposition 2.7 (assuming the constants in the linear time algorithm do not depend on genus).

2.3. **General Applications.** From (2), we quickly see that K_v is genus-n circular, where

$$n = \left\lceil \frac{(v-2)(v-3)}{12} \right\rceil \tag{3}$$

This follows from the directly from the proof of Lemma 2.4.

On a slightly different topic, it is very interesting to note that, in a sense, K_5 is $\Delta - Y$ equivalent to the graph $K_{3,3} \cup e(2,6)$, where $K_{3,3}$ is the complete bipartite graph with six nodes. (see figure 6).

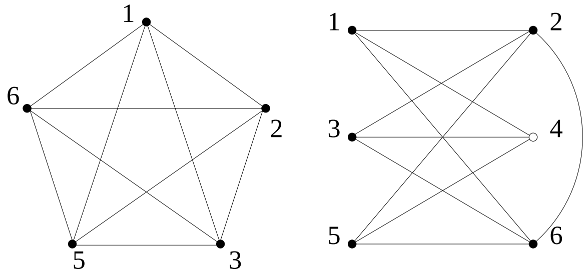

FIGURE 6. K_5 and its $\Delta - Y$ Equivalent, $K_{3,3} \cup e(2,4)$

This is interesting in the context of the following definition and theorem.

Definition 2.9. A graph G' is a *subdivision* of a graph G if G' is obtained from G by replacing edges of G with series connections.

Theorem 2.10. *(Kuratowski Reduction Theorem) A graph G is non-planar iff it is a supergraph of some subdivision of K_5 or $K_{3,3}$.*

For this reason, K_5 and $K_{3,3}$ are known as the *forbidden minors* (also known as Kuratowski Graphs) for the plane. From figure 6, we now see that the two forbidden minors for genus-0 are practically $Y - \Delta$ equivalent to one another. Can we extend this to other genera? First, we consider a theorem about forbidden minors.

Theorem 2.11. *(Robertson-Seymour Theorem) For any genus g, the list of forbidden minors for that genus is finite.*

Unfortunately, not much more is known. Besides the plane, a complete list of forbidden minors in known only for the projective plane, which has 103 ([8]). The torus is known to have at least one thousand ([9]). So, even if we can make a guess about relations between forbidden minors due to $Y - \Delta$s (and, as we'll see, $\bigstar - K$'s), it will be impossible, at this point, to check.

One way to generalize the fact about the relation between K_5 and $K_{3,3}$ is as follows.

Proposition 2.12. *For $n \geq 3$, K_{2n-1} is $\bigstar - K$ equivalent to a supergraph of $K_{n,n}$.*

Proof. We will only consider the $n = 8$ case, from which it should be clear in general. Consider an arbitrary circular embedding of K_7 (see figure 7).

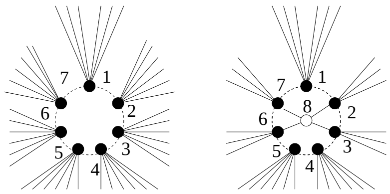

FIGURE 7. An Arbitrary Circular Embedding of K_7, and a $\bigstar - K$'d K_7 Embedded in a Similar Fashion.

Then do a $\bigstar - K$ on nodes the K_4 with vertices v_2, v_3, v_6, and v_7. Place the $\bigstar$ inside the circle. Call the new interior node v_8. Then we now have a supergraph of the bipartite graph $K_{4,4}$, with one part being v_2, v_3, v_6, and v_7, and the other part being v_1, v_4, v_5, and v_8. $\square$

It is worth noting that the genus of the circular embedding of K_7 (e.g. a standard embedding of K_8) by going to $K_{4,4}$ (in other words, given a topological embedding of K_7, we found a topological embedding of $K_{4,4}$; this is called an "embedding extension problem", or EEP, and is discussed in [1]). This is an interesting way to put a bound on genus—specifically, that the genus of K_8 is at least the genus of $K_{4,4}$, or, in general, the genus of $K_{n,n}$ is at most the genus K_{2n}. Now, certainly, this is an obvious fact. Indeed, even the idea of trying to put bounds on the genus of complete and bipartite graphs is pointless, as the genus of such graphs is well known ([3]). Nonetheless, the method of proof is very interesting. Perhaps, by considering multiple "boundary circles", we could use a similar argument to put bounds on other types of graphs, such as certain generalized forms of bipartite graphs.

In light of the previous discussion, we make the following statement.

Conjecture 2.13. For an arbitrary genus g, every forbidden minor is a $\bigstar - K$ equivalent to a complete graph, a complete bipartite graph, or the connected sum of forbidden minors of lower genera (to be sure, we must define what exactly we mean by connected sum, but that will not be discussed in this paper).

If true, this could potentially be quite useful, as it may allow us to find an exact formula for the number of forbidden minors of an arbitrary genus, as well as give a constructive method to find all such forbidden minors. Even if not true, it should give, inductively, a large class of forbidden minors for a given genus.

It is worth noting that not all $Y - \Delta$ equivalent graphs necessarily have the same genus.

Conjecture 2.14. There exists a graph G such that any $Y - \Delta$ changes the genus of G.

This question is important because, if true, it shows just how much a medial graph can change under the slightest transformation.

A motivation for this statement comes from a simple example. Consider a Δ embedded around a loop on the torus. Then doing a $\Delta - Y$, we can't put the new interior vertex "inside" the torus—we must put it on the surface. Depending on how the three nodes in the Δ were connected to other nodes, this could force a change in the genus. I am pretty sure that this conjecture is true; an example that should work is shown in figure 7, which is genus-1 because it contains a supergraph of $K_{3,3}$.

One could check the given graph proves conjecture 2.14 by doing a $\Delta - Y$ and using the program provided in the appendix on the new graph to see if the it is genus-2 (or higher).

This conjecture raises further questions. How much can the genus of a graph change under a $Y - \Delta$? Under a general $\bigstar - K$?

2.4. **A Simple Program.** The algorithm for finding a combinatorial embedding of an arbitrary graph in linear time is very complicated, and would take a while to

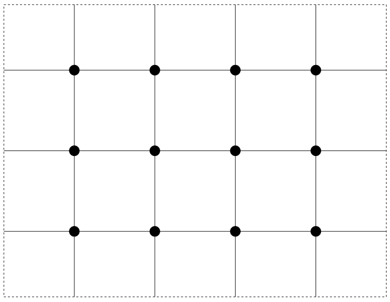

FIGURE 8. A Graph Such That Any $\Delta - Y$ Changes the Genus?

implement (however, it could be a very interesting future REU project to attempt to do so). Proposed here is a very simple, albeit phenomenally slow algorithm to determine the genus of a graph. This algorithm is implemented as a Matlab program in Appendix A.

In the 18th Century, Euler and Descartes independently proved a very important result, which, as we will see, has applications to graph theory.

Theorem 2.15. *(Descartes-Euler Polyhedral Formula) Suppose P is a polyhedron of genus zero. Let v be the number of vertices of P, E be the number of edges, and F be the number of faces. Then*

$$V - E + F = 2 \tag{4}$$

A generalized version of this equation was proved by Poincare.

Theorem 2.16. *(Poincare Formula)*

$$V - E + F = \chi(g) \tag{5}$$

where $\chi(g) = 2 - 2g$ is the Euler Characteristic of the polyhedron, and g is the genus of the polyhedron.

Before we make use of this theorem, we need one definition.

Definition 2.17. Let G be a simple graph, where $d(v_i)$ is the degree of v_i. A *face of a combinatorial embedding of G* is a sequence $k_1, k_2, ...k_n$ such that for all $1 \leq i \leq n$, v_{k_i} is adjacent to $v_{k_{(i-1)modn}}$, as well as $v_{k_{(i \bmod n)+1}}$ (here we define $0 \bmod n$ to be n). Additionally, we require that if $a_{k_i}[m] = v_{k_{(i-1) \bmod n}}$, then $a_{k_i}[(m+1) \bmod d(v_i)] = v_{k_{(i \bmod m)+1}}$. In other words, if we have a coordinate embedding of G consistent with the combinatorial embedding of G, then a *face of G* is a sequence $k_1, k_2, ...k_n$ so that the polygon whose edges are formed by the cycle $e(v_{k_1}, v_{k_2}), e(v_{k_2}, v_{k_3}), ...e(v_{k_{n-1}}, v_{k_n}), e(v_{k_n}, v_{k_1})$ contains no edges.

Now, suppose we have a combinatorial embedding of an arbitrary graph. Then, identify the vertices of the graph with vertices of a polyhedron, the edges of the graph with the edges of a polyhedron, and the faces of a graph with the faces of a polyhedron. Determine the genus of the polyhedron using Poincare's Formula, then that will be the genus of combinatorial embedding of the graph.

Further justification to appear.

Since we know there are a finite (though very large) number of combinatorial embeddings of a given graph, we can use Poincare's Formula to determine the genus of that graph by checking the genus of all possible combinatorial embeddings. Furthermore, we can find the n such that a graph G with boundary is genus-n circular by making use of Lemma 2.8.

3. Medial Graphs

3.1. **Background.** In this paper, we assume the convention for drawing medial graphs as defined in [4]. He proved that, in general, medial graphs are two colorable; in particular, that we could actually draw them. We examine properties of medial graphs in an attempt to determine if there is an (easy) way to generalize the statement in the circular planar case from [7], namely that a circular planar graph is recoverable iff its medial graph has a lens.

We consider particularly symmetric graphs here, as it makes the medial graph drawing process simpler. As noted in (1), there are many, many different ways to embed graphs. We will say two representations of medial graphs are *equivalent* if they came from the same combinatorial embedding of a graph. (1) then gives us an idea of how many different medial graphs we can have for a given graph. This makes recovering information from medial graphs extremely difficult.

3.2. **Examples.** Rather than choosing random combinatorial embeddings of graphs, we can look at some specific graphs individually. We will try to always take an circular embedding that is of the smallest genus possible. Figure 9 shows one way we can find a genus-1 circular embedding of an arbitrary annular graph.

By using this method, we get a very simple, symmetric medial graph. Note the difference between the medial graph of $G(3, 2)$ in figure 9 and that of [4], which is extremely difficult to interpret. Figure 9's embedding quickly generalizes to other annular graphs of the form $G(n, m)$.

Is there any obvious geometrical difference between the medial of $G(3, 2)$, known to not be recoverable, and $G(4, 2)$, known to be recoverable? Figure 10 shows the medial graph of $G(4, 2)$.

Other than some extra regions due to the increased number of nodes and edges, there are no differences between the two medial graphs whatsoever. This is in stark contrast to the circular planar case.

Consider another example, shown in figure 11.

Using the same strategy as for the previous annular graphs, we can find a highly symmetric embedding of the square-in-square graph on the torus. This could potentially yield interesting results since the square-in-square is neither infinite to one nor one to one—it is two to one ([10]). When we draw the medial graph, something very interesting is immediately clear: there is a circle in the medial graph around the boundary circle! Unfortunately, this seems to say nothing about recoverability properties of the graph, as figure 11 shows. By moving some of the "bars" around, we get a medial graph corresponding to a recoverable graph (use $\bigstar - K$). By adding another circle around the boundary circle, we find an infinite to one graph (parallel connections).

As one last example, we'll consider the triangle in triangle in triangle graph shown in figure 12. By arguments made in [10], we find that it is four to one.

As we have seen, it is extremely important to have a symmetric embedding of the circular graph if we are to reasonably make any sense of the medial graph. Figure 12 shows a method we could use to find such an embedding.

It is important to realize that the method shown will not always work. In the case shown, we "shrunk" the boundary circle across holes of the genus-3 surface to get a circle without any edges inside. Depending on the combinatorial embedding, it may be impossible to do this. However, it should always work in the n-gon-in-n-gon-in-n-gon case to give a surface of genus n.

Additional information to appear.

3.3. **Strategies For Drawing Medial Graphs.** We list some strategies to make visual representations here, as well as ways to make those representations as simple as possible. For quick reference, some information is repeated from above.

- Use the method of joining boundary circles to find circular embeddings of graphs, as shown in figures 9 and 12. Always try to maintain as much symmetry as possible.
- Figure 12's method of cutting apart surfaces of high genus can be useful in some cases, but another helpful strategy, given an embedding on a flattened representation of a surface, is to maintain the combinatorial representation on the planar representation. Remember that depending on the directions hidden surfaces face in a flattened represenation, one may have to reverse the ordering of edges around nodes in the planar representation.
- Always try to examine embeddings on surfaces of the lowest genus, for simplicity. If you're really stuck, you could use Lemma 2.8 in conjunction with a slightly modified version of appendix A to find a minimal genus embedding of a circular graph.

3.4. **Obtaining Information From Medial Graphs.** Needless to say, the examples shown are extremely dissapointing. First, they provide counterexamples to the Conjecture 4.1 of [4]. Second, though great care was taken to draw the medial graphs in such a way so that there properties were most obvious, nothing is very clear. There are no simple iff conditions as in the circular planar case.
All is not lost, however. The lens property is certainly extendable from the circular planar case if we make the right restrictions. As long as we have a lens entirely contained in a region homeomorphic to the disc, we can apply the same argument to see that the graph is not recoverable. Furthermore, it seems likely that all non-recoverable graphs contains some type of lens, or, conversely, no recoverable graph contains lenses of a certain type (see [4] for examples of what these lenses might be, such as bubbles or (linked) non-region-bounding lenses). It could be fruitful to examine the medial graphs of several combinatorial embeddings of the same graph and search for a property invariant relative to such embeddings. Alternatively, one could consider medial graphs of embeddings that are non-circular, such as those mentioned in [**?**]

Another way we might be able to get recoverability information from medial graphs counting certain properties. Counts were useful, as we say in Theorem 2.16, for determining the genus of a combinatorial embedding of a graph. Furthermore, counts were extremely useful in the Cut-Point Lemma from [7], which was very important in recoverability. For convenience, it is reproduced here.

Theorem 3.1. *(Curtis and Morrow, Cut-Point Lemma) Suppose A is a finite family of chords in the disc, and assume that A is lensless. Let X and Y be a pair of cut-points for A. With $n(X,Y)$, $m(X,Y)$, and $r(X,Y)$ defined as above,*

$$m(X,Y) + r(X,Y) - n(X,Y) = 0$$

There is a prononounced similarity between Theorem 2.16 and Theorem 3.1, namely, they are both of the form $A + B + C = D$. It seems unlikely that there is just a coincidence. Perhaps there is a generalized Cut-Point Lemma for non-planar graphs that relates cut-points to the genus.

If, as it may be, there is a way to generalize the Cut-Point Lemma, how could we make analagous recovery arguments for non-planar graphs? The theorem for recoverability for circular planar graphs depended on the idea of *well-connected graphs G_k*, graphs with the property that, as discussed by [7], for every circular pair of sequences of boundary nodes $(P;Q) = (p_1, p_2, ...p_k; q_1, q_2, ...q_k)$, there is a k-connection from P to Q. Certainly, this doesn't make much sense in the non-planar case—the Jordan Curve Theorem no longer applies, so it doesn't make much sense to talk about circular pairs. However, there may be other ways to extend what we mean by well connected graphs. Perhaps there is a different type of "circular." One could attack this problem by considering z-sequences of medial graphs (see [**?**]). In particular, a planar graph G_k is well connected iff its z-sequence is of the form $1, 2, ..k, 1, 2, ...k$. By defining other, non-planar graphs in terms of their z-sequences, we may be able to generalize the idea of well-connected graphs.

One final possibility for gaining information from medial graphs would be to consider them in matrix form. Choose some arbitrary geodesic, and order the remaining geodesics in order around the boundary circle (so that we have the z-sequence). Then, for define a matrix $M = (m_{ij})$ with width equal to twice the number of boundary nodes and height equal to the maximum number of times any single geodesic is intersected. Let the ith column of M correspond to the ith element of the z-sequence, so that (m_{ij}) is the jth geodesic to intersect the ith geodesic. If j is greater than the number of times geodesic i is intersected, define (m_{ij}) to be zero. Note that the columns of M are only determined up to cyclic permutations of any z-sequence. The matrix below is an example of such a numerical representation of the medial graph given in figure 12, where the z-sequence starting from the top and proceeding clockwise is $\{1, 2, 3, 4, 5, 6, 1, 4, 5, 2, 3, 6\}$.

$$\begin{bmatrix} 2 & 1 & 4 & 3 & 6 & 5 & 4 & 1 & 2 & 5 & 6 & 3 \\ 4 & 5 & 6 & 1 & 2 & 3 & 2 & 3 & 6 & 1 & 4 & 5 \\ 6 & 3 & 2 & 5 & 4 & 1 & 6 & 5 & 4 & 3 & 2 & 1 \\ 2 & 1 & 4 & 3 & 6 & 5 & 4 & 1 & 2 & 5 & 6 & 3 \\ 4 & 5 & 6 & 1 & 2 & 3 & 2 & 3 & 6 & 1 & 4 & 5 \end{bmatrix}$$

In the circular planar case, we can see very quickly from the medial graph whether or not there is a lens—we just check to see if there is a column in which the same nonzero number occurs twice. However, there is a lot more information than that! What can we deduce about graphs from this medial matrix? Note that, as defined here, the matrix will not suffice to represent all types of medial graphs. We must decide on a good convention for representing bubbles (see figure 11) as well.

4. Conclusion

So far, medial graphs in the non-planar case have been disappointing. As the previous discussion suggests, however, all is not lost. Most likely, any information that comes from medial graphs will be of a form similar to the Cut-Point Lemma and the Poincare Formula. At this point, though, nothing is clear.

5. Further Research

Here are some topics for further research. Most are discussed in detail above.

- Medial graphs on non-orientable surfaces
- Amalgamating non-planar medial graphs (see [11])
- Generalizing well-connected graphs
- Generalizing the Cut-Point Lemma
- Medial graphs in the non-circular case
- An implementation of [1]'s embedding algorithm
- Examining matrix representations of medial graphs
- Ideally, combining some of the above items to solve the inverse problem in general

Appendix A. A Matlab Program For Determining the Genus of a Graph

The program consists of three files:

- genus.m
- genusoce.m
- permute.m
- possibilities.m

The main file is genus.m, which the user calls to test for the genus of the graph. genus.m takes an $n*n$ symmetric adjacency matrix $M = (m_{ij})$ such that $m_{ij} = 1$ if v_i is adjacent to v_j, and $m_{ij} = 0$ if v_i is not adjacent to v_j. It is assumed that this graph contains no loops. Failure to place entries correctly in the adjacency matrix will result in error. The program does not check that the adjacency matrix is valid when it starts; instead, an error will occur during computations.

A.1. genus.m.

Note: Not all K's in the following paragraph are the same. Sorry for the confusion, this will be fixed.

The function genus(K) takes an adjacency matrix and creates an arbitrary combinatorial embedding. This embedding is represented as an $n*m$ matrix $M = (m_{ij})$, where n is the number of nodes and m is the largest valence of any node. The ith row of M corresponds to a_i as defined in 2.5. m_{ij} is zero if j is greater than the number of elements of a_i. Then, genus(K) uses possibilities(K) to determine how many combinatorial embeddings of K there are. It then executes a for loop that cycles through every possible permutation of the rows of M, and uses genusoce(K) to calculate the genus of each of those combinatorial embeddings. Finally, it returns the minimum genus, as well as the number of combinatorial embeddings that were of that genus.

```
function g = genus(K);
```

```
maxvalence = 0;
vert = size(K,1);
horiz = size(K,2);

for(i = 1: vert)
    counter = 0;
    for(j = 1: horiz)
        if K(i,j) ~= 0
            counter = counter + 1;
        end
    end
    if counter > maxvalence
        maxvalence = counter;
    end
end

L = zeros(vert, maxvalence);

for(i = 1: vert)
    counter = 1;
    for(j = 1: horiz)
        if K(i,j) ~= 0
            L(i, counter) = j;
            counter = counter + 1;
        end
    end
end

stopcount = possibilities(K);
mingenus = genusoce(L);
totalmin = 0;

for (i = 0: stopcount - 1)
    G = L;
    gen = genusoce(permutate(i,G));
    if gen == mingenus
        totalmin = totalmin + 1;
    end
    if gen < mingenus
        mingenus = gen;
        totalmin = 1;
    end
end

disp(stopcount - 1);
disp('Minimum Genus:')
disp(mingenus);
disp('Possible Ways to Embed in That Genus:')
```

```
disp(totalmin);
g = mingenus;
```

A.2. **genusoce.m.** genusoce(K) takes the matrix M as defined in subsection A.1. Then, it counts the number of vertices by determining the vertical size of M, and the edges by counting the number of nonzero entries in M and dividing by two (since each edge ends on two vertices). Finally, it calculates the number of faces as follows:

(1) Define a new zero matrix C with the same dimensions as K.
Look at each element of (k_{ij}) of K. If it is zero or is nonzero in C, skip to (5). Otherwise, continue.
(2) Call $k(ij)$ the starting entry, and the current entry.
(3) Go to the vertex given by the current entry (e.g. go that row of M). Search that row until i is found. Call the entry to the right of the one equaling i (or the first entry if the entry equal to i is the last non-zero entry in the row) current vertex.
(4) If the current vertex is not the starting vertex, jump to (3).
(5) Repeat until all entries of K have been considered, or equivalently, C = K.

```
function g = genusoce(K)

faces = 0;
vertices = size(K,1);
maxvalence = size(K,2);
edges = 0;

for(i = 1: vertices)
    for(j = 1: maxvalence)
        if K(i,j) ~= 0
            edges = edges + 1;
        end
    end
end

edges = edges/2;
C = zeros(vertices,maxvalence);

for(i = 1: vertices)
    for(j = 1: maxvalence)
        if K(i, j) ~=0 && C(i, j) == 0
            prev = 0;
            faces = faces + 1;
            startvertex = i;
            C(i, j) = K(i, j);
            t = j - 1;
            if t == 0
                t = maxvalence;
```

```
            end
            while K(i, t) == 0
                t = t - 1;
            end
            prev = K(i, t);
            currentvertex = i;
            loopvertex = K(i,j);
            while ~(loopvertex == startvertex && prev == currentvertex)
                counter = 1;
                while K(loopvertex,counter) ~= currentvertex
                    counter = counter + 1;
                end
                currentvertex = loopvertex;
                counter = mod(counter, maxvalence) + 1;
                while K(currentvertex, counter) == 0
                    counter = mod(counter, maxvalence) + 1;
                end
                C(currentvertex, counter) = K(currentvertex, counter);
                loopvertex = K(currentvertex, counter);
            end
        end
    end
end

g = (2 - vertices + edges - faces)/2;
```

A.3. **permutate.m.** permutate(K,i) takes as input a matrix M as defined in subsection A.1, as well as an integer i such that $0 \leq i \leq \textit{possibilities}(L) - 1$ where L is the original adjacency matrix. Then, using an ordering of all possible ways to permute the nonzero entries of M in each row, it takes the ith element of that ordering (assuming that the ordering is indexed from zero to possibilities(L)). It then returns that element. This ordering is illustrated by the following example.

Consider the ordered set 1,2,3,4. We want to have an ordered list of all possible permutations of this set, except for those that are equivalent via cyclic permutation to a permutation already on the list. Then the list is as follows:

(0) $\{1, 2, 3, 4\}$

(1) $\{1, 2, 4, 3\}$

(2) $\{1, 3, 2, 4\}$

(3) $\{1, 3, 4, 2\}$

(4) $\{1, 4, 2, 3\}$

(5) $\{1, 4, 3, 2\}$

where the number in parenthesis is the number of the element on the list, corresponding to the integer i. If $i = 0$, do nothing. If $i = 1$, swap the last two digits. If $3! < i \leq k * (2!)$, swap the third element from the right with the $(3 - k)$th element from the right. Subtract $k * (2!)$ from i, and repeat the process until $i = 0$.

This shows how we can order the cyclic permutations of a set. If number of nonzero elements is more than four, then add additional if statements with 4 and 3 replacing 3 and 2, and so on.

Now we can extend this idea to a matrix. If the first row has n nonzero elements, and we are given $i > (n - 1)!$, call permutate on M with the first row removed, and as the input integer send $i' = \left\lfloor \frac{i}{(n-1)!} \right\rfloor$. Then, on the matrix M' that is returned, execute permutate with $i'' = i \bmod (n - 1)!$. We now have an ordering to the list of ways to permute the individual rows of a matrix so that now two permutations are equal after a cyclic permutation.

```
function g = permutate(i,G);

vert = size(G,1);
horiz = size(G,2);
lastentry = horiz;

while(G(1,lastentry) == 0)
    lastentry = lastentry - 1;
end

if i >= factorial(lastentry - 1)
    H = G;
    r = floor(i / factorial(lastentry - 1));
    i = mod(i, factorial(lastentry - 1));
    G(2:vert,1:horiz) = permutate(r,H(2:vert,1:horiz));
end

j = 1;

while i >= factorial(j);
    j = j + 1;
end

k = 1;

while i >= k*factorial(j - 1)
    k = k + 1;
end

k = k - 1;
```

```
a = G(1, horiz + 1 - j + k);
G(1, horiz + 1 - j + k) = G(1, horiz + 1 - j);
G(1, horiz + 1 - j) = a;
i = i - k * factorial(j - 1);

if i ~= 0
    G = permutate(i,G);
end

g = G;
```

A.4. **possibilities.m.** This function takes an adjacency matrix A and returns the number of combinatorial embeddings of that matrix, as given by (1).

```
function g = possibilities(K)

t = 1;

for(i = 1: size(K,1))
    s = 0;
    for(j = 1: size(K,2))
        s = s+K(i,j);
    end
    t = t*factorial(s-1);
end

g = t;
```

References

[1] Bojan Mohar. "A Linear Time Algorithm for Embedding Graphs in an Arbitrary Surface," SIAM Journal of Discrete Math. 12 (1999) 6-26.
[2] Simon Pai. "Equivalence of Planarity and Circular Planarity" REU 2003.
[3] Mathworld. "Graph Genus". http://mathworld.wolfram.com/GraphGenus.html
[4] Tim DeVries. "Extending the Medial Graph to Non-Circular Planar Resistor Networks." REU 2003.
[5] William S. Massey. "Algebraic Topology: An Introduction." Harcourt, Brace and World, Inc. 1967.
[6] Thomas Carlson. "A Natural Representation for a Class of Circular Planar Graphs." REU 1999.
[7] Edward B. Curtis and James A. Morrow. "Inverse Problems for Electrical Networks." Series on Applied Mathematics – Volume 13. World Scientific, 2000. W
[8] Jonathon L. Gross, Thomas W. Tucker. "Topological Graph Theory." John Wiley and Sons, Inc., 1987.
[9] Bojan Mohar, Carsten Thomassen. "Graphs on Surfaces." Johns Hopkins University Press, 2001.
[10] Jennifer French and Jerry Pan. "2^n to 1 Networks." REU 2004.
[11] Brian Lehmann. "Recoverability of Spliced Networks." REU 2002.

University of Washington,
E-mail address: nwr@u.washington.edu

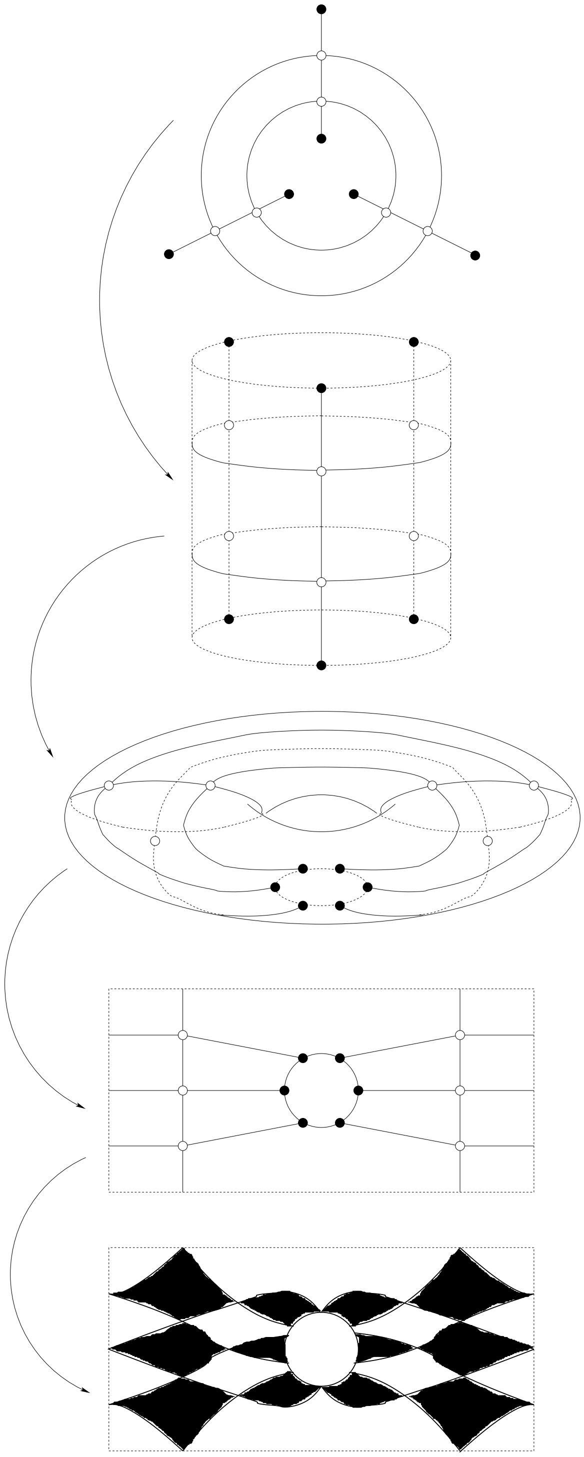

FIGURE 9. Steps To Draw a Highly Symmetric Medial Graph of $G(3, 2)$.

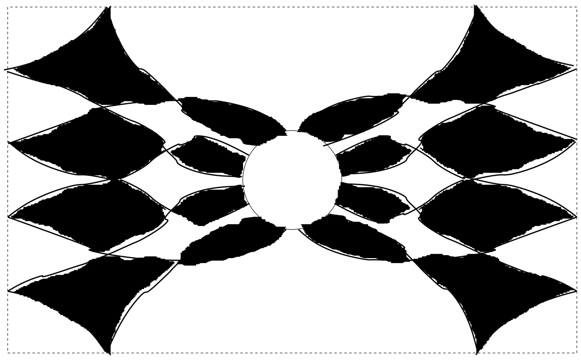

FIGURE 10. An Medial Graph of An Embedding of $G(4, 2)$ Similar to the Given Embedding of $G(3, 2)$. Notice There Are No Geometrical Differences Other Than What We Would Expect.

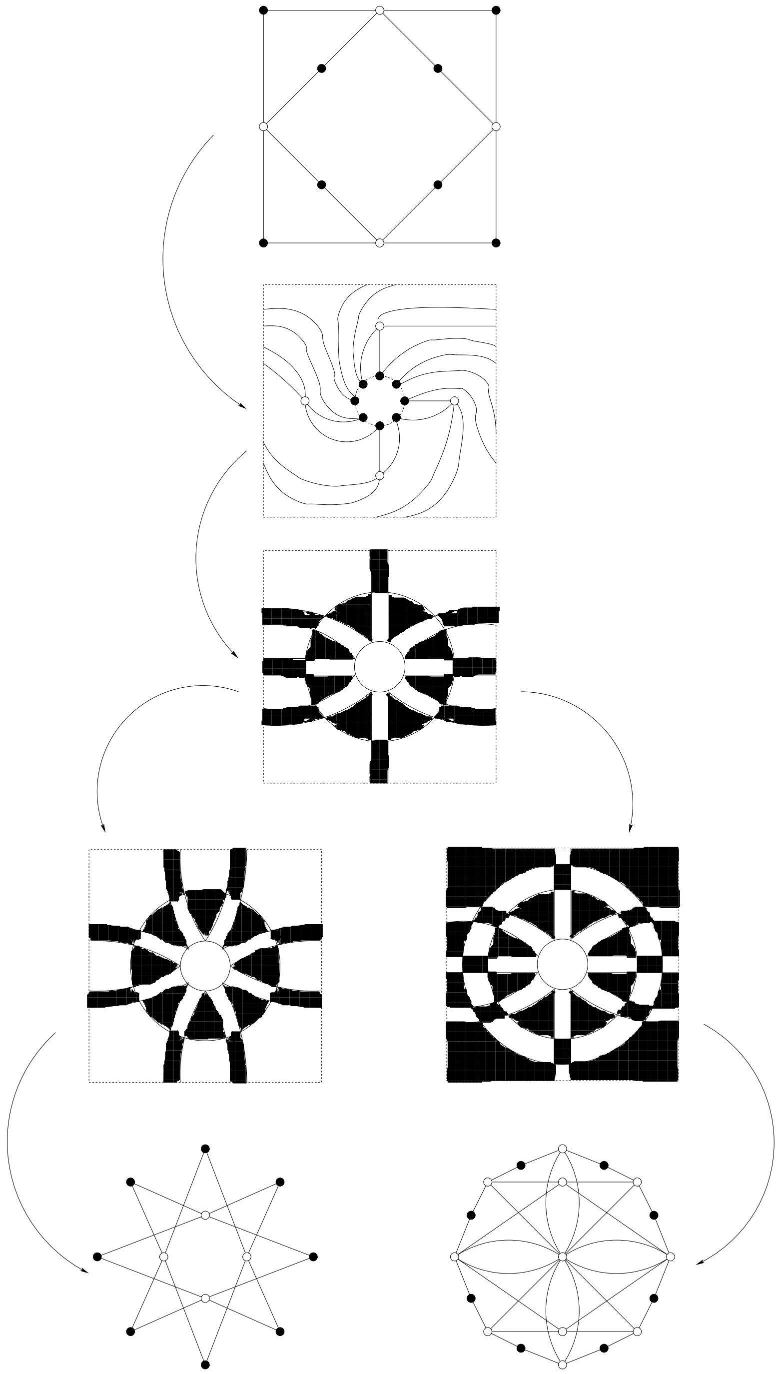

FIGURE 11. Three Graphs With Different Recoverability Properties. Clockwise From Top: Two to One Graph, Infinite to One Graph, and One to One Graph.

FIGURE 12. To Appear.