Summary: These notes cover the eighth week of classes. We present a theorem on the additivity of graph genera. Then we move on to cycles of embedded graphs, the three-path property, and an algorithm for finding a shortest cycle in a family of cycles of a graph.

1 Additivity of Genus

Theorem 1.1 (Additivity of Genus). Let $G$ be a connected graph, and let $B_1, B_2, \ldots, B_r$ be the blocks of $G$. Then the genus of $G$ is,

$$g(G) = \sum_{i=1}^{r} g(B_i)$$

and the Euler genus of $G$ is,

$$eg(G) = \sum_{i=1}^{r} eg(B_i).$$

Observation 1.2. In order to prove the theorem it suffices to prove that if $G = G_1 \cup G_2$, where $G_1 \cap G_2 = \{v\}$, then $g(G) = g(G_1) + g(G_2)$ and $eg(G) = eg(G_1) + eg(G_2)$.

Proof. We set

$$g(G) = \min\{g(\Pi) \mid \Pi \text{ is an orientable embedding of } G\}$$

$$g(G_1) = \min\{g(\Pi_1) \mid \Pi_1 \text{ is an orientable embedding of } G_1\}$$

$$g(G_2) = \min\{g(\Pi_2) \mid \Pi_2 \text{ is an orientable embedding of } G_2\}$$

Let $v_1, v_2$ be vertices in $G_1, G_2$ respectively, distinct from $v$. Note that the local rotations of $v_1, v_2$ are independent, but the local rotation of $v$ is not. So if we take $\Pi$ an embedding of $G$ and split it along $v$ we obtain $\Pi_1, \Pi_2$ embeddings of $G_1, G_2$ respectively. Facial walks in $G$ that do not pass from $G_1$ to $G_2$ remain unchanged in $\Pi_1, \Pi_2$, however if a facial walk does pass from $G_1$ to $G_2$ then when we split we form an extra face. But we also have an extra vertex. So, $g(\Pi) = g(\Pi_1) + g(\Pi_2)$. 

* Lecture Notes for a course given by Bojan Mohar at the Simon Fraser University, Winter 2006.
Claim: The min value of \( g(G) \) is attained at an embedding where the local rotation at the cut vertex \( v \) groups the edges from \( G_1 \) into a single block.

Assume that this is not the case. Then each facial walk must use “mixed angles” in pairs (by mixed angle we refer to the angle formed at \( v \) between two edges, one from \( G_1 \) and one from \( G_2 \)). By considering the local rotations we see that we must traverse an odd number of negative signatures from \( v \) to \( v \) on any walk using mixed angles. Thus we have a 1-sided cycle in our embedding. However this is a contradiction as we assumed that \( \Pi \) was orientable. This proves the claim. It also proves that

\[
g(G) = g(G_1) + g(G_2).
\]

This proof can be repeated with minor changes to prove that,

\[
eg g(G) = \neg g(G_1) + \neg g(G_2).
\]

This completes the proof. \(\square\)

## 2 Induced Embeddings

Given \( \Pi \) an embedding of \( G \) we wish to develop the notion of a corresponding embedding \( \Pi' \) of \( H \) a subgraph of \( G \). Suppose that \( e \in E(G) \) is not a cut-edge of \( G \), then we can consider deleting \( e \) from \( G \). Without loss of generality, we may assume that \( \lambda(e) = +1 \). If \( e \) is in two distinct facial walks, then we can see that \( G - e \) embeds in the same surface as \( G \), this follows as \( G - e \) will have the same Euler characteristic as \( G \). If \( e \) appears twice on one face, then \( G - e \) will embed in a simpler surface (the Euler genus will decrease by either 1 or 2). From this we can see that in the specific case of multigraphs, the embedding of the underlying simple graph cannot have larger genus, and never changes from orientable to non-orientable.

## 3 Cycles of Embedded Graphs

We start by giving an intuitive explanation of the classes of cycles we will consider. For example, say we are considering the surface \( S_3 \), pictured below.

- \( C_1 \) bounds a disk on the surface. Such cycles are called contractible cycles.
- Cutting along \( C_2 \) separates the surface. Such cycles are called surface separating cycles.
• $C_3$ and $C_4$ are called surface non-separating cycles.

• Note that contractible cycles are also surface separating.

Now we formally define these notions. Let $C$ be a two-sided cycle of a $\Pi$-embedded graph $G$. We may assume that $\lambda(e) = +1$ for all edges $e \in E(C)$. We can arbitrarily choose a “perspective” from which to view $C$. Now since $C$ is two-sided we have a “left” and “right” side of $C$ with respect to our perspective. This allows us to define:

$$
E_L := \text{the set of all edges incident to a vertex of } C \text{ embedded on the “left” of } C
$$

$$
E_R := \text{the set of all edges incident to a vertex of } C \text{ embedded on the “right” of } C
$$

$$
G_L(C) := \text{the subgraph of } G \text{ consisting of } E_L \text{ and all vertices and edges in } G - C \text{ reachable from } E_L
$$

$$
G_R(C) := \text{the subgraph of } G \text{ consisting of } E_R \text{ and all vertices and edges in } G - C \text{ reachable from } E_R
$$

**Definition 3.1.** For a two-sided cycle $C$, if $G_L(C) \cap G_R(C) \subseteq C$, then we say that $C$ is a surface separating cycle.

**Proposition 3.2.** Let $C$ be a surface separation cycle of a $\Pi$-embedded graph $G$. Consider the induced embeddings of $G'_L := G_L(C) \cup C$ and $G'_R := G_R(C) \cup C$. The sum of the Euler genera of the embeddings of $G'_L$ and $G'_R$ is $eg(\Pi)$.

**Proof.** Claim #1: Every facial walk of $G$ is either a facial walk of $G'_L$ or a facial walk of $G'_R$.
If a facial walk never reaches $C$, the claim holds trivially. Suppose we have a facial walk $F$ that intersects $C$. Since we take all signatures on $C$ to be positive and all rotations to be equal, we can see that each time $F$ enters $C$ on a right edge, it will leave on a right edge (and vice versa). This proves the claim.

Claim #2: $C$ is facial in $G'_L$ and $G'_R$.
This holds for the same reason as Claim #1.
Thus the number of faces of $G$ is two less than the sum of the number of faces in $G'_L$ and $G'_R$. Now we simply apply Euler’s formula to $G, G'_L$ and $G'_R$ to prove the proposition. \( \square \)

**Definition 3.3.** A cycle $C$ of a $\Pi$-embedded graph $G$ is $\Pi$-contractible if it is surface separating and the Euler genus of either the induced embedding of $G_L(C) \cup C$ or $G_R(C) \cup C$ is zero.

Note that this is equivalent to $C$ bounding a disk on the surface.

**Definition 3.4.** If $C$ is contractible and $G_L(C) \cup C$ has genus zero, then

$$
\text{int}(C) = \text{int}(C, \Pi) := G_L(C)
$$

$$
\text{Int}(C) = \text{Int}(C, \Pi) := G_L(C) \cup C
$$

Now we can classify cycles of embeddings as:
4 Cutting Surfaces Along Cycles

Cutting a surface along a cycle \( C \) gives rise to a graph in which \( C \) is replaced by 2 cycles, \( C' \) and \( C'' \) (both are copies of \( C \)). The edges on the left of \( C \) (with respect to some perspective) are incident with the corresponding vertices of \( C' \), the vertices on the right of \( C \) are incident with the corresponding vertices of \( C'' \). The graph that we obtain by performing this operation is isomorphic to \( G_L(C) \cup G_R(C) \cup C' \cup C'' \). Embeddings of \( G \) induce embeddings of \( C' \) and \( C'' \) as one might expect.

**Proposition 4.1.** If \( C \) is surface separating, then cutting along \( C \) gives two graphs isomorphic to \( G' \) and \( G'' \) respectively, and \( eg(G'_L, \Pi) + eg(G'_R, \Pi) = eg(G, \Pi) \).

If \( C \) is two-sided, but not surface separating, then the graph obtained after cutting along \( C \), \( G' \), is connected, and \( eg(G', \Pi) = eg(G, \Pi) - 2 \).

If \( C \) is one-sided, then the graph \( G' \), obtained after cutting along \( C \), is connected, and \( eg(G', \Pi) = eg(G, \Pi) - 1 \).

Note that in proposition 4.1, the orientability may change from non-orientable to orientable in the last two cases.

In the following drawings of the Projective Plane, Torus, and Klein Bottle we have the following:

- \( C_1, C_2, C_4 \) and \( C_7 \) are contractible
- \( C_3, C_9 \) and \( C_{10} \) are one-sided
- \( C_5 \) and \( C_6 \) are non-contractible
- \( C_8 \) is surface separating and non-contractible
- \( C_{11} \) is two-sided and non-separating
Note that the only surface separating cycles on the Torus are contractible, as the Torus is orientable. Also note that cutting along $C_1$ is the inverse operation of adding a twisted handle.

5 The Three-Path Property

Given vertices $x, y$ of a graph $G$ and internally disjoint $xy$-paths $P_1, P_2, \ldots, P_r$, we denote the cycle formed by paths $P_i, P_j$ as $C_{ij}$.

**Definition 5.1.** Let $\mathcal{C}$ be a family of cycles in $G$. $\mathcal{C}$ has the **three-path property** if:

\[ \forall x, y \in V(G), \forall P_1, P_2, P_3 \text{ internally disjoint} \ xy\text{-paths}, \text{ if } C_{12} \notin \mathcal{C} \text{ and } C_{23} \notin \mathcal{C}, \text{ then } C_{13} \notin \mathcal{C}. \]

**Example 5.2.** The following are examples of families with the three-path property:

1. $\mathcal{C} = \{C | \text{the length of } C \text{ is odd}\}$
2. $\mathcal{C} = \{\text{cycles with and odd number of edges in } E'\}, \text{ where } E' \subseteq E(G)$
3. $\mathcal{C} = \{\text{one-sided cycles of } \Pi\}$
4. $\mathcal{C} = \{\text{non-contractible cycles of } \Pi\}$

We now give a short proof of 4.

**Proof.** Take $x, y \in V(G)$ and $P_1, P_2, P_3$ internally disjoint $xy$-paths. Assume that $C_{12}$ and $C_{23}$ are contractible. First we alter $\Pi$ so that all signatures on $C_{12}$ are positive. If $P_3 \subseteq \text{Int}(C_{23})$ then the result is clear. Similarly, if $P_1 \subseteq \text{Int}(C_{23})$ then the result is also clear. So we need only consider the case where $P_2$ lies “between” $P_1$ and $P_3$. Now we have that $C_{13}$ is surface separating and $\text{int}(C_{13}) = \text{int}(C_{23}) \cup \text{int}(C_{12}) \cup P_2$. It follows from Euler’s formula that the genus of $\text{Int}(C_{13})$ is zero, and thus $C_{13}$ is contractible. This completes the proof.

We now present an algorithm for finding a shortest cycle in $\mathcal{C}$, where $\mathcal{C}$ has the three-path property.
Algorithm 5.3. Input: A graph $G$ and a family of cycles $C$ with the three-path property.
For all $v \in V(G)$:
   Build the breadth-first search spanning tree of $G$ starting at $v$, $T_v$.
   For every edge $e \notin E(T_v)$:
      Let $C_e$ be the unique cycle in $T_v + e$.
      Choose a shortest of the cycles $C_e$ to be $C_v$.
Choose $C$ to be a shortest of the cycles $C_v$.
Return: $C$ is a shortest cycle in $C$.

Proposition 5.4. Algorithm 5.3 correctly finds a shortest member of $C$ in time $O(nqT + nq)$, where $n = |G|$, $q = \|G\|$, and $T$ is the time complexity of $C \in C$ queries.

Proof. For a vertex $v \in V(G)$ let $T_v$ be the BFS tree built by Algorithm 5.3. Let $C \in C$, $C = v_0, \ldots, v_{k-1}$, and $v_0 \in V(C)$ be selected subject to $C$ being shortest in $C$ and then having minimum intersection with $T = T_{v_0}$. We prove that Algorithm 5.3 finds a cycle of length $|C|$.

Claim 1: $d_G(v_0v_i) = d_C(v_0v_i)$ for $i = 0, \ldots, k - 1$. Let $i$ be smallest such that $d_G(v_0v_i) \neq d_C(v_0v_i)$ and let $P$ be the shortest path from $v_0$ to $v_i$. $P$ contains a subpath $P'$ that connects two vertices $x, y \in V(C)$. Let $A, B$ be the cycles, formed by $P'$ and $xCy, yCx$. $A$ and $B$ are shorter than $C$, thus none of them is in $C$. By the three-path-property, neither is $C$. This contradiction establishes the claim.

Claim 2: There exists $e \in E(C)$, incident with $v_t$, $t = \lfloor \frac{k}{2} \rfloor$, such that $C - e \subseteq T$. Let $i$ be smallest such that $v_iv_{i+1} \notin E(T)$. By symmetry we may assume $i < t$. Let $P$ be the path from $v_0$ to $v_{i+1}$ in $T$. The claim follows by the same argument as the previous one, using the fact that $P$ has length $i + 1$, implied by Claim 1.

Since there is only one edge of $C$ missing in $T$, this cycle is examined by the algorithm, thus the cycle that is chosen at $v_0$ and subsequently in $G$ has length at most $|C|$.

BFS tree can be found in time $O(q)$ and using it the length of the cycles can be compared in constant time. Thus there is at most $O(qT + q)$ time spent in the loop for each of the $n$ vertices. The complexity follows. \hfill $\square$

Corollary 5.5. If $C$ satisfies the three-path property and membership in $C$ can be determined in polynomial time, then the above algorithm finds a shortest cycle in $C$ in polynomial time.

Note that Algorithm 5.3 can be applied to find a shortest one-sided cycle, non-contractible cycle, surface non-separating cycle, etc.