Math 137a	Professor: Padraic Bartlett

Lecture 3: Trees and Art Galleries

Week 2	*UCSB 2014*

1 Cut-Edges and Spanning Trees

So: last class, we showed the following conditions were all equivalent ways to define what a tree is:

1. G is connected and has no cycles.
2. G is connected and has $n-1$ edges.
3. There exists exactly one path between any two vertices in G.

We mentioned that two other conditions were equivalent to these three. I want to start today by returning to one of those two conditions:

4. G is connected, and every edge of G is a cut-edge[1].

Specifically: I realized that I want us to actually show that this condition is equivalent to being a tree, because we need it today! To do this, we merely need to prove the following lemma:

Lemma 1. *An edge $e = \{u, v\}$ of a graph G is a cut-edge iff it doesn't belong to any cycle.*

Proof. Take any edge $e = \{u, v\}$. Remove this edge from our graph: if the graph is still connected, then there is some path from u to v not involving e; consequently, if we add e to the end of this path, we get a cycle. Thus, if e is not a cut-edge, it's involved in a cycle.

Conversely: suppose that $e = \{u, v\}$ lies in a cycle. Let P be the path from u to v that doesn't use e (i.e. go the other way around the cycle.) Pick any x, y in G; because G is connected, there's a path from x to y in G. Take this path, and edit it as follows: whenever the edge e shows up, replace this with the path P (or P traced backwards, as needed.) This then creates a walk from x to y; by deleting cycles, this walk will always become a path, and thus G is connected. So if e is involved in a cycle, it's not a cut-edge. $\square$

Corollary 2. *G is a tree if and only if it's connected and all of its edges are cut-edges.*

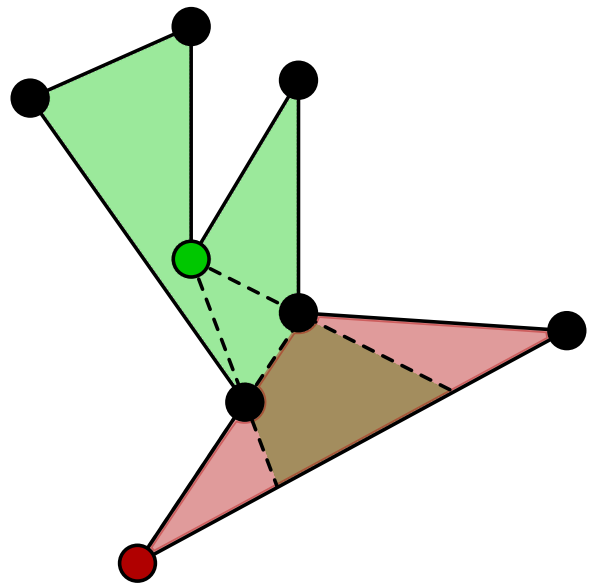
Proof. By the above lemma, G is connected and all of its edges are cut-edges if and only if G is connected and acyclic, which we know to be equivalent to being a tree. $\square$

[1] A **cut-edge** of a connected graph is an edge e such that if e is deleted from our graph, our graph is disconnected.

2 The Art Gallery Problem

Ok! Tree prelude over. Now I can state the main goal of this set of lecture notes: the art gallery problem! Consider the following question:

Question 3. *Suppose that you have an art gallery that is shaped like some sort of n-polygon, and you want to place cameras with $360°$-viewing angles along the vertices of your polygon in such a way that the entire gallery is under surveillance. How many cameras do you need?*



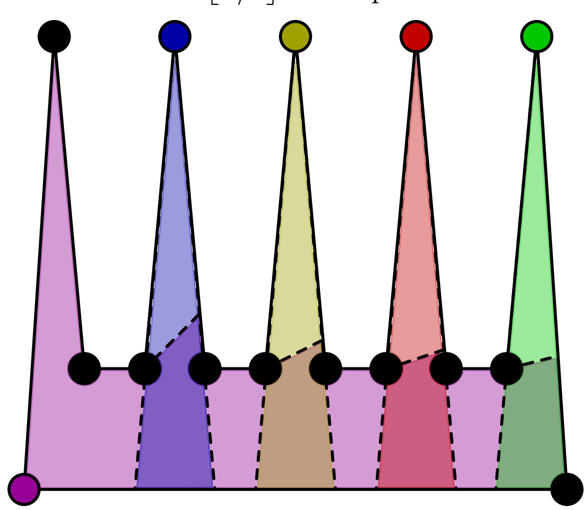
A gallery guarded by 2 guards, Red and Green.

One trivial upper bound you can come up with is n guards: just put one guard on each vertex of our polygon with n sides!

Can we do better? As it turns out, we can!

Claim. (Chvátal) You need at most $\lfloor n/3 \rfloor$-many cameras to guard a n-polygon.

It bears noting that this bound of $\lfloor n/3 \rfloor$ is sharp. Consider the following art gallery:



A crown-shaped art gallery.

In the above sort-of "crown-shaped" art gallery, each prong of the crown (i.e. triangle) needs to have a guard on one of its three vertices to guard the entire triangle, as no other vertices can "see" the entirety of that prong. Therefore, you need one guard for each prong; i.e. $n/3$ guards, for a crown with $n/3$ prongs (i.e. n vertices.)

To prove Chvátal's theorem, we need a few lemmas first:

Lemma 4. *If G is a n-polygon with $n \geq 4$, then there is some line segment formed by two of the vertices in G that lies entirely in G.*

Proof. Let v be the leftmost vertex of G. (If there is a tie, take v to be the top leftmost vertex of G.) Let u and w be v's neighbors, and examine the line segment $\overline{uw}$. If this lies entirely in G, great! Otherwise, it must cross some edge of G; consequently, there must be a vertex of G that lies inside of the triangle spanned by the three points u, v, w. Let x be the vertex furthest from the line segment $\overline{uw}$ that lies in this triangle. Then, look at the line segment $\overline{vx}$; because x is the furthest point in Δuvw from $\overline{uw}$, there can't be any edges of G that are crossed by this line segment (as one of their endpoints would necessarily be closer to v.) So $\overline{vx}$ lies entirely in G. $\square$

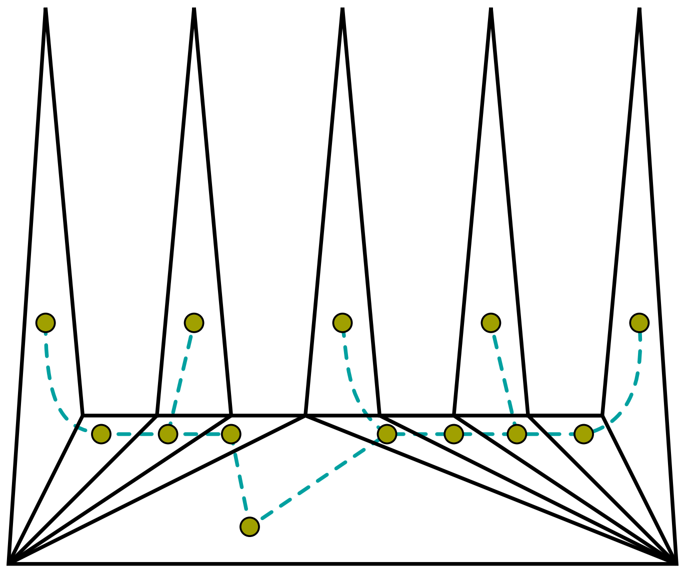
Corollary 5. *Any n-polygon can be divided into $n - 2$-triangles.*

Proof. Using the process above, repeatedly divide our n-polygon into a pair of smaller polygons, one with k vertices and the other with $n+2-k$ vertices, until all of these polygons are triangles. By induction, it is not hard to see that the number of these triangles is $n-2$. $\square$

So: we can turn any polygon into a number of triangles stuck to each other! We use this to turn any art gallery on n vertices into a **graph** on $n - 2$ vertices, as follows:

- Start by taking our polygon G and turning it into a collection $\{T_i\}_{i=1}^{n-2}$ of triangles.
- For each triangle T_i, associate a vertex t_i.
- Connect two vertices t_i, t_j with an edge if their corresponding triangles T_i, T_j share a face.

Call this graph T' the **dual graph** of T.



Turning the crown into a tree.

This is a graph! Furthermore, it's a fairly special kind of graph: it's a tree! We prove this here:

Lemma 6. *Let G be a polygon, T be a triangulation of G performed as above, and let T' be the dual graph to this triangulation (i.e. put a vertex in the center of every face of T , and connect two faces iff they share an edge.) This graph is a tree.*

Proof. Let T be our triangulated polygon. In our construction above, each of the edges of T' corresponds to a diagonal of the polygon G, that divides our polygon into two distinct smaller polygons. Because cutting our polygon G along one of those diagonals will always divide the polygon into two disconnected pieces, doing so will always result in two triangles that are no longer connected by a chain of triangles with adjacent faces!

In other words: in the dual graph T' that we made above, removing any edge disconnects our graph! Consequently, we know that in the dual graph T' of T, every remaining edge of T' is a cut-edge. Therefore, as we showed earlier in lecture, T' must be a tree. $\square$

This tree is remarkably useful; in particular, we can use its structure to create a system for assigning guards! We do this here:

Lemma 7. *Take a polygon G that has been triangulated as described earlier. Then we can color each of the vertices of G either red, blue or green, so that each triangle contains one vertex of each color.*

Proof. For our triangulated polygon G, take the dual graph/tree T' that we constructed above, and pick some vertex t_0 in it. For all relevant integers k, let T'_k be the collection of vertices that are distance k away from v, for every k. (The **distance** of two vertices from each other is the length of the shortest path between them.)

Color the vertices of T as follows:

- Take the triangle in G associated to t_0 – i.e. the only element in T'_0 – and color its three vertices red, green and blue.

- Suppose we've colored all of the vertices attached to triangles with corresponding vertices in T'_i, for some i. Now, look at the triangles corresponding to vertices in T'_{i+1}. Each triangle associated to a vertex t_{i+1} in this set shares exactly one edge with some triangle associated to a vertex in T'_i; this is because if our vertex is distance $i+1$ from t_0, then (by taking the path of distance $i+1$ and walking one step closer to t_0) there is an adjacent vertex (and thus face-sharing triangle) at distance i, i.e. in T'_i.

  Furthermore, because T' is a tree, there is exactly one edge from any t_{i+1} to vertices in the set $\bigcup_{j=0}^{i+1} T'_j$. This is because the existence of any other edge would create a cycle, because it would give us two distinct paths to t_0! (Two different paths for the two different ways to get to vertices in some T_j, and then from there we can get to t_0 by taking the unique path from that t_j vertex to t_0.)

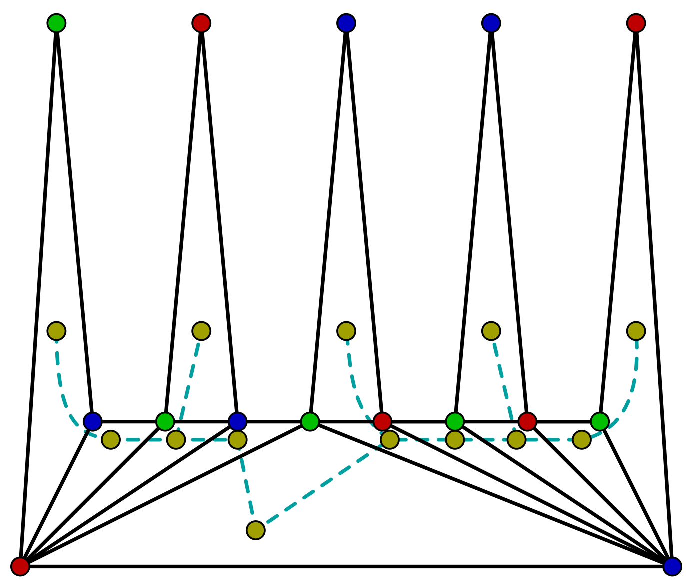
  Therefore, the triangle associated to t_{i+1} shares a face with **only one other** triangle in all of the sets that we've already colored! Therefore, only two of its vertices have been assigned colors. Thus, there is always some spare third color to use to color its third vertex! Use this to color its third vertex, and repeat for all vertices in T_{i+1}.

- Repeat this process until every vertex in T is colored. By construction, every triangle has one vertex of each color.

$\square$

Corollary 8. *You need at most $\lfloor n/3 \rfloor$-many cameras to guard a n-polygon.*

Proof. By the above, create a triangulation and 3-coloring of our polygon G with the colors $\{R, G, B\}$. Now, station guards at whichever color is used the least number of times in this triangulation! Each guard can see everything in their assigned triangle, as there are no walls of our art gallery (i.e. lines of our polygon) that intersect these triangles, by construction. Therefore, the entire art gallery is guarded. $\square$



To guard this crown, simply pick one of (red, green, blue,) and station guards at vertices of that color.