GRAPH COLORING

ABY SOUMARE

ABSTRACT. In this laboratory, we will be studying the topics of graph coloring. More specifically, the number of distinct colorings a given graph may have. In this report, we will present examples of different kinds of graphs, their specific coloring algorithms, and multiple Theorems we discovered along the way.

1. Introduction

First, we must understand the topic of graph coloring. The topic of graph coloring is determining the number of colors needed to color all vertices of a graph in such a way that adjacent vertices do not share the same color. The question of graph coloring explores this: Does there exist a solution to the function from the vertices of a graph to a set of colors where,

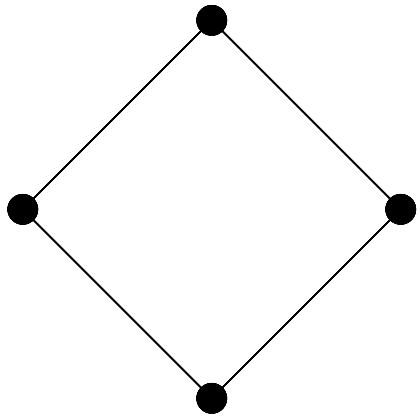
(1) There are enough colors to color each vertex.
(2) No two vertices that share an edge are colored in the same way.

See the formal definition of a graph:

Definition 1.1. A graph is a set of vertices $v = \{v_1, v_2, ..., v_n\}$ and set of edges $e = \{e_1 = v_1v_2, e_2 = v_2v_3, ..., e_n = v_rv_n\}$ where $e_1 = v_1v_2$ represents the edge between the vertices v_1, v_2.

Example 1.2. Diagram showcasing a graph:

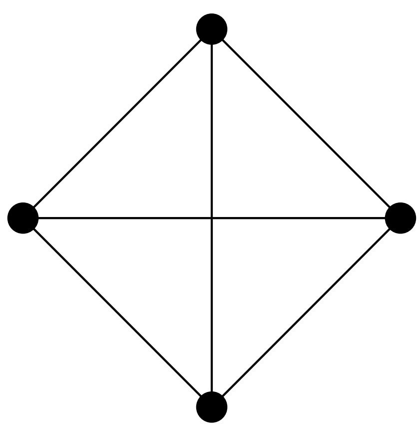
I thank Professor Robinson and my classmates in Math 251 at Mount Holyoke College for helping me complete this report.



In this graph, we can see that there is no direction, there are no loops, and there are no double edges. A graph without these properties is referred to as a **simple graph**. However, that definition will not be needed for the purposes of this report. A kind of graph that will pertain to this report is a complete graph. See the definition here:

Definition 1.3. A complete graph, G, is a graph in which all vertices are connected to one another via an edge.

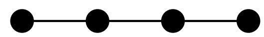
Example 1.4. Example of a complete graph:



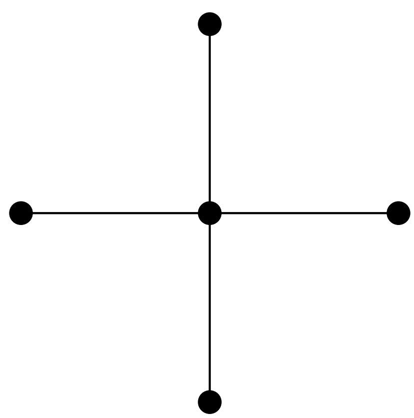
Notice how every vertex is connected to every other vertex with an edge. We will dive deeper into the way in which to color this graph later on in this report. Another kind of graph we will be observing is a star graph. See this definition:

Definition 1.5. An n-path graph is a graph with n vertices and $n-1$ edges.

Example 1.6. An example of an n-path graph:



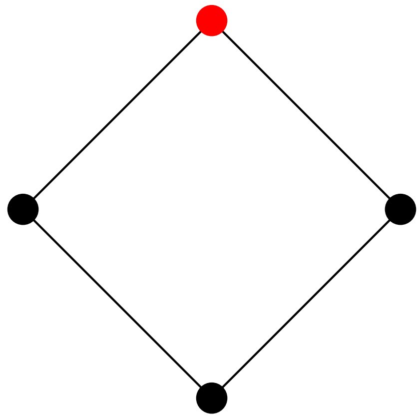
Example 2 of an n-path graph:



See how in the second example there are 5 vertices and 4 different edges. Remember this definition, as this is what we are referring to when we mention n-path graphs later on in this report. Let us define our final kind of graph here:

Definition 1.7. A cycle graph is a graph that contains a closed loop where a given vertex, v_1, is connected to a vertex, v_2, which is connected to a vertex, to a vertex, to a vertex v_n, that is then connected to v_1.

Example 1.8. Example of a cycle graph:

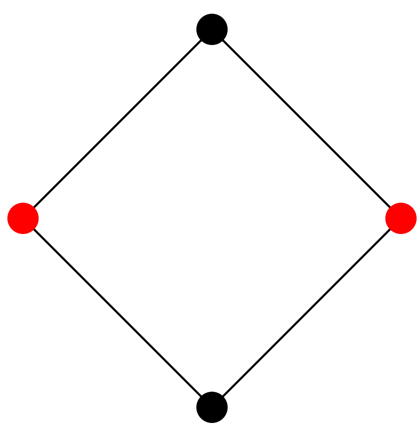


Notice how the red vertex, v_1, is connected to the vertex to the left, v_2, which is connected to the vertex below it, v_3, which is connected to the vertex to the right of it, v_4, which is finally connected to v_1.

Now, we will get more specific about our language in terms of graph coloring. See this definition for a proper coloring of a graph:

Definition 1.9. A proper coloring for a graph, **G**, is a function from the vertices of a graph, $v \in G$, to a set of colors such that no v_1, v_2 have the same color if they share an edge, $e = v_1 v_2$.

Example 1.10. Diagram showcasing a proper coloring for a graph:



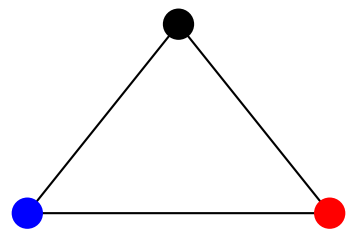
See how in this graph every vertex that shares an edge with another vertex has a different color from the one it is connected to. So, in order to properly color this graph, we can see that only 2 colors are needed. More colors can be utilized to color the graph, but we know the minimum amount would be 2. Let us formally define this here:

Definition 1.11. The minimum number of colors needed to properly color a graph is referred to as **the chromatic number**.

So, given a specified number of colors, how can we determine the total number of ways to color a graph such that no two adjacent vertices are the same color? This is the graph coloring counting problem, and we will delve into this question in more detail in the rest of this lab, with multiple examples of different kinds of graphs.

2. Analysis

Consider the graph of an equilateral triangle G:



Notice, for this graph to be properly colored, we need 3 different colors. If we were to only have 2 colors, two vertices would have the same color and also share an edge. Now that we know the chromatic number for a triangle such as this is 3, we must also ask ourselves how many different ways can we color this graph? We know that we have 3 different colors we can use and 3 vertices to color. So, let us walk through this problem step by step.

(1) First, take a vertex and color it any color.
(2) Then, take the second vertex and color it any color but the first color.
(3) Finally, take the last vertex and color it your last color.

To summarize, the first vertex has 3 different ways to color it, the second has 2 different ways to color it, and the last has only 1 way to color it. Multiplying these numbers together, we have that this graph has 6 different ways to properly color it.

Instead of going through every graph and individually counting the different ways we can color the graph, we want to utilize a function that will solve this problem for us. In order to do that, we must first define a term:

Definition 2.1. A chromatic polynomial for a graph, G, is denoted as $P_G(x)$. This is a polynomial of $x = \#ofcolors$ that evaluates the number of proper colorings of a G.

So, we can see that there is an equation that shows us the number of different ways we can color the triangle graph we were observing above. That equation is:

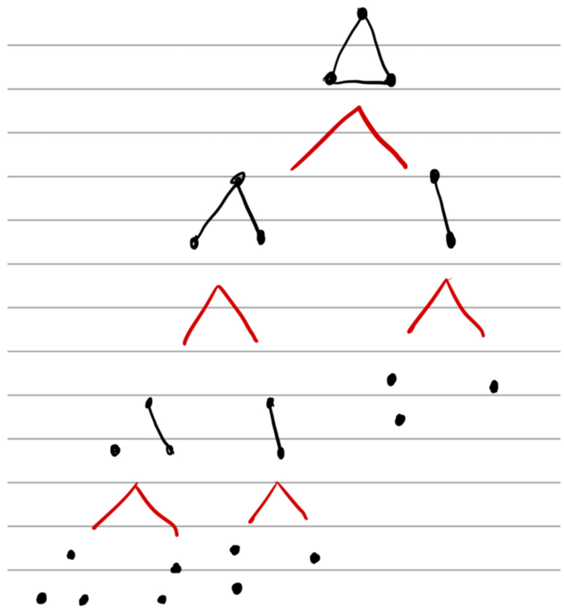
$$P_G(x) = x(x-1)(x-2).$$

Since the first vertex has x colors it can be, the second has $x-1$ colors it can be, and the third has $x-2$ colors it can be. With the chromatic number $x = 3$, we know that our $P_G(x) = 3(2)(1) = 6$. This is the same as the number we got from counting each vertex individually, too.

Now, we need a method in which to create our chromatic polynomial for any given graph. In order to do this, we will be using Birkhoff-Lewis' Algorithm. See the definition here:

Definition 2.2. The Birkhoff-Lewis Algorithm is an algorithm that derives a graph's chromatic polynomial. It works by simplifying a graph's polynomial into two chromatic polynomials of two related graphs. We then continue to simplify the two graphs into smaller related graphs until we are left with only vertices.

Example 2.3. Example showcasing the Birkhoff-Lewis Algorithm:



Notice how moving left will always subtract an edge, while moving right will combine two vertices together. This can be written as $P_G(x) = P_{G-e}(x) - P_{G/e}(x)$. What this means is that we are counting the graph sans edges and dividing out the extra vertices we are counting. The Birkhoff-Lewis Algorithm helps us evaluate very large and complicated graphs we might not have the chromatic polynomial for yet.

Now we are going to begin with our findings from labs.

Conjecture 2.4. The terms of $P_G(x)$, with no cycles, increase in absolute value and then decrease in absolute value in such a way where:

(1) If n, the number of vertices, is even, then the number of increasing terms and decreasing terms is equal to $\frac{n}{2}$.

(2) If n is odd, then the number of increasing terms is the ceiling (e.x. the ceiling of $2.2 = 3$, and the ceiling of $2.7 = 3$) of $\frac{n}{2}$, and the number of decreasing terms is the floor (e.x. the floor of $2.2 = 2$, and the floor of $2.7 = 2$) of $\frac{n}{2}$.

You can see the pattern displayed in this table showcasing the terms on the xs of a $P_G(x)$:

Degree of $P_G(x)$	Number of n	Increasing terms	Decreasing terms
2	2	1	1
3	3	2	1
4	4	2	2
5	5	3	2
6	6	3	3
7	7	4	3
8	8	4	4
9	5	5	4

Now, we do not yet have the tools to prove this, so it will stay as a conjecture. However, there are many findings within this lab that we are able to prove. Let us start with this Theorem:

Theorem 2.5. *Suppose a graph G with n vertices, the chromatic polynomial will always begin with the leading term x^n.*

Proof. If we think about this in a combinatorial way, assume we have a graph $P_{G_n}(x)$ with n vertices and with no edges. If we were to color these vertices with x colors, then our $P_{G_n}(x) = x^n$. We know this for certain because no vertex is connected to another vertex via an edge, so each n vertex can be colored any x color.

Now, consider a graph, G, with n vertices, x colors to choose from, and $e \geq 1$ edges. The first vertex we choose to color will have x^n options to color it since that is our starting point. The second, third,..., and nth vertex we choose will have $x - 1$, $x - 1$,..., $x - n$ colors to choose from respectively.

Finally, via the Birkhoff-Lewis algorithm, we are deleting vertices of degree one and their edges without changing their chromatic polynomial, since we are not changing the leading term. The leading term, x^n, corresponds to the total number of vertices in the graph, so we know deleting degree one vertices will not change a graph's polynomial. Therefore, we know that for any polynomial with n vertices, we will always be leading with x^n □

After proving the behavior of the vertices of a graph, we also want to say something about the edges. See this Theorem:

Theorem 2.6. *The coefficient of x^{n-1} in the chromatic polynomial of any graph G with n vertices is $-e$ where e is the number of edges in G.*

Proof. We will proceed via induction on e. For our base case, we suppose $e = 0$, then there are n vertices in G. Our chromatic polynomial would then be x^n. This result is precisely what we want, and we have therefore proven our base case. We will now continue by assuming our hypothesis.

Suppose a graph G with n vertices and $e \geq 0$, then our chromatic polynomial will be $P_G(x) = x^n - ex^{n-1}...$ is true. We want to show that this applies for $e+1$ edges for our inductive step. So, we want to prove $P_G(x) = x^n - (e+1)x^{n-1}...$ is true. We can utilize the Birkhoff-Lewis Algorithm to state that:

$$P_G(x) = P'_G(x) - P_G"(x).$$

We know that $P'_G(x)$ has n vertices and e edges, and $P_G"(x)$ has $n-1$ vertices and less than e edges, though we don't know the exact number of $e \in P_G"(x)$. Continuing our inductive hypothesis, we can say that:

$$P'_G(x) = x^n - ex^{n-1}....$$

We also want to make a statement about $P_G"(x)$:

$$P_G"(x) = x^{n-1} - ex^{n-2}....$$

By the Birkhoff-Lewis Algorithm, we will continue by subtracting these two polynomials from each other. See that in action here:

$$P_G(x) = (x^n - ex^{n-1}...) - (x^{n-1} - ex^{n-2}...).$$

This reduces down to:

$$P_G(x) = x^n - (e+1)x^{n-1}....$$

Now, we have our desired outcome $(e+1)$. Therefore, we have proven our Theorem via induction on the edges. $\square$

After proving this, we now want to move on to proving the polynomials to the graphs previously defined. We will start with the polynomial for an n-path graph:

Theorem 2.7. *The formula for the chromatic polynomial for an n-path graph P_G is $P_G(x) = x(x-1)^{n-1}$ for $n \geq 1$.*

Proof. We will be proving this Theorem using induction on the vertices. Our base case is when $n = 1$. Consider the equation:

$$P_{G_1}(x) = x(x-1)^{1-1}.$$

This will simplify down to just x, which is our single vertex with no edges. This is precisely what we want. So, we can assume our hypothesis holds and move on to our inductive step.

We assume $P_{G_n}(x) = x(x-1)^{n-1}$ for $n \geq 1$ is true, and we want to show that $P_{G_{n+1}}(x) = x(x-1)^n$ for $n+1 \geq 1$ is also true. Consider the Birkhoff-Lewis Algorithm applied to our $P_{G_{n+1}}(x)$:

$$P_{G_{n+1}}(x) = xP_{G_n}(x) - P_{G_n}(x).$$

Substituting the equation in for $P_{G_n}(x)$:

$$P_{G_{n+1}}(x) = x * x(x-1)^{n-1} - x(x-1)^{n-1}.$$

Then combine the $(x-1)^{n-1}$ to get:

$$P_{G_{n+1}}(x) = x(x-1)^n.$$

This is precisely the equation we want. Therefore, we have proven our Theorem via induction on the vertices and the Birkhoff-Lewis Algorithm. $\square$

After that, we want to create a polynomial for a complete graph. For this, we have Theorem 2.8 which states:

Theorem 2.8. *The chromatic polynomial formula for a complete graph with n vertices G_n is $P_{G_n}(x) = x(x-1)(x-2)...(x-(n-1))$ for $n \geq 1$.*

Proof. We will prove our Theorem via induction. For our base case, we have 1 vertex. Our equation will then be $x(x-1)^0 = x$, which is what we want. So, we know our base case holds and we can continue with assuming our hypothesis.

Suppose the chromatic polynomial for a complete graph is $P_G(x) = x(x-1)(x-2)...(x-(n-1))$ is true, we want to show that our inductive step, $P_{G_{n+1}}(x) = x(x-1)(x-2)...(x-(n-1))(x-n)$, is also true. In order to prove our inductive step, we will be using Birkhoff-Lewis Algorithm. Given a complete graph with $n+1$ vertices, the algorithm states that the chromatic polynomial is:

$$P_{G_{n+1}}(x) = P_{G'_{n+1}}(x) - P_{G''_{n+1}}(x).$$

By the Birkhoff-Lewis Algorithm, we can deduce that $P_{G''_{n+1}}(x) = P_{G_n}(x)$ for a complete graph because the added vertex, v_{n+1}, once combined with another vertex in $P_{G''_{n+1}}(x)$ will simply bring us back to the graph $P_{G_n}(x)$, which is a graph covered by our hypothesis. So, now we can simplify our earlier equality to be:

$$P_{G_{n+1}}(x) = P_{G'_{n+1}}(x) - P_{G_n}(x).$$

Now we will apply the Birkhoff-Lewis Algorithm to $P_{G'_{n+1}}(x)$, which is simply $P_{G_n}(x)$ with $n-1$ edges connected to the new vertex v_{n+1}

because we deleted an edge in our last use of the Birkhoff-Lewis Algorithm. We want to assign new notation to make this proof easier to digest. Consider $P_{G'_{n+1}}(x) = P_{H_{n+1}}(x)$. Now consider the Birkhoff-Lewis Algorithm applied to $P_{H_{n+1}}(x)$:

$$P_{H_{n+1}}(x) = P_{H'_{n+1}}(x) - P_{H''_{n+1}}(x).$$

Again, we know that $P_{H''_{n+1}} = P_G(x)$ by the same reasoning as above. So we will again apply the Birkhoff-Lewis Algorithm to $P_{H'_{n+1}}$.
After applying the Birkhoff-Lewis Algorithm n number of times, we will be left with:

$$P_{G_{n+1}} = xP_G(x) - nP_G(x).$$

After deleting all edges connecting the new vertex v_{n+1} to the graph $P_G(x)$, we know we can color that vertex any color, x, possible. That is the reason we are getting $xP_G(x)$ in our equation. A similar reasoning can be used to explain why we have $nP_G(x)$, but for $nP_G(x)$, we were deleting the edges n times.
Now, we can factor out the $P_G(x)$ to get:

$$P_{G_{n+1}} = P_G(x)(x - n).$$

This equality is precisely the equation we want. Therefore, we have proven our Theorem using induction and Birkhoff-Lewis Algorithm. □

Finally, we want to create a polynomial for a cycle graph. See our final Theorem 2.9 and its proof here:

Theorem 2.9. *The chromatic polynomial formula for a cycle graph* G_n *is* $P_{G_n} = (x-1)^n + (-1)^n(x-1)$ *for* $n \geq 2$.

Proof. We will prove our Theorem via induction on the vertices. Our base case is when $n = 2$. Consider:

$$P_{G_2}(x) = (x-1)^2 + (-1)^2(x-1).$$

Since $-1^2 = 1$, then we know $P_{G_2}(x)$ now equals:

$$P_{G_2}(x) = (x-1)^2 + (x-1).$$

Now expand $(x-1)^2$:

$$P_{G_2}(x) = (x-1)(x-1) + (x-1).$$

Multiply out $(x-1)(x-1)$ to get:

$$P_{G_2}(x) = x^2 - x - x + 1 + (x-1).$$

We simplify down and remove parentheses:

$$P_{G_2}(x) = x^2 - 2x + 1 + x - 1.$$

Simplify even further to get:

$$P_{G_2}(x) = x^2 - x.$$

Factor out an x:

$$P_{G_2}(x) = x(x-1).$$

This is precisely what we want for our base case. So, we can assume that $P_{G_n}(x) = (x-1)^n + (-1)^n(x-1)$ is true as our hypothesis, and we now want to prove that $P_{G_{n+1}}(x) = (x-1)^{n+1} + (-1)^{n+1}(x-1)$ is true for our inductive step to hold. In order to prove this, consider the Birkhoff-Lewis Algorithm on our equation for $P_{G_{n+1}}(x)$:

$$P_{G_{n+1}}(x) = x(x-1)^n - ((x-1)^n + (-1)^n(x-1)).$$

We know this is true via Birkhoff-Lewis and our Theorem 2.8 which uses the algorithm in more detail. After multiplying out the negative, $P_{G_{n+1}}(x)$ turns into:

$$P_{G_{n+1}}(x) = x(x-1)^n - (x-1)^n - (-1)^n(x-1).$$

Then we factor out $(x-1)^n$ to get:

$$P_{G_{n+1}}(x) = (x-1)^{n+1} + (-1)^{n+1}(x-1).$$

As you can see, this is precisely the form we want. Therefore, we have proven our Theorem via induction on the vertices and the Birkhoff-Lewis Algorithm. $\square$

3. Conclusion

In this lab, we went studied the topic of graph coloring, what it means to properly color a graph, and polynomials to derive the number of proper colorings for 4 different kinds of graphs. Questions that may arise from this report could be:

(1) What about the polynomials of graphs we have not yet studied?
(2) What more can be said about the terms on the xs of a polynomial?
(3) What is a good practical use of the graph coloring problem?

While the majority of these questions cannot be answered within this report, the last question in particular can. Graph coloring is a very useful tool to aid us in many ways. One way that this can be especially useful in is in helping us solve scheduling problems. An example presented in the textbook from class[1] is:
Imagine you are the department head for the math department at your college, and you have surveyed your students, asking them what classes they want to take for next semester. You want to use the information given to you in the survey to ensure you are not scheduling classes in such a way where a student can't take all their desired classes. Instead of taking the approach where you put all of this information into a table or any other means, we can simply make a graph with all the different classes as their own vertex, and draw a line connecting two vertices (connecting two classes) if a student wants to take both of them. If we can properly color the graph, then it is possible for you to schedule all classes in such a way where all students can take their desired classes. This is a very practical use of the graph coloring problem, and you are now well-equipped to understand other real-world applications.

References

[1] Department of Mathematics and Statistics at Mount Holyoke College, *Laboratories in Mathematical Exploration: A Bridge to Higher Mathematics*, Springer-Verlag, New York, 1997.

Email address: souma23a@mtholyoke.edu

Department of Mathematics and Statistics, Mount Holyoke College, South Hadley, MA 01075