CSCI 5451 Fall 2015

Week 8 Notes

Professor Ellen Gethner

October 4, 2015

History of the Four Color Problem

- Frances Guthrie, a British lawyer and botanist was coloring a map of England and thought he had discovered a method to color an arbitrary map using at most four colors (1853).

History of the Four Color Problem

- Frances Guthrie, a British lawyer and botanist was coloring a map of England and thought he had discovered a method to color an arbitrary map using at most four colors (1853).
- Guthrie's conjecture was fallaciously proved by Alfred Kempe, a barrister, in 1878.

History of the Four Color Problem

- Frances Guthrie, a British lawyer and botanist was coloring a map of England and thought he had discovered a method to color an arbitrary map using at most four colors (1853).
- Guthrie's conjecture was fallaciously proved by Alfred Kempe, a barrister, in 1878.
- In 1890, Percy Heawood (a mathematician) found an irreparable flaw in Kempe's proof.

History of the Four Color Problem

- Frances Guthrie, a British lawyer and botanist was coloring a map of England and thought he had discovered a method to color an arbitrary map using at most four colors (1853).
- Guthrie's conjecture was fallaciously proved by Alfred Kempe, a barrister, in 1878.
- In 1890, Percy Heawood (a mathematician) found an irreparable flaw in Kempe's proof.
- The Four Color Conjecture was solved in the affirmative in 1976 by Appel, Haken, and Koch by way of joining together theory and computation; the latter took hundreds of hours of computation time.

History of the Four Color Problem

- Frances Guthrie, a British lawyer and botanist was coloring a map of England and thought he had discovered a method to color an arbitrary map using at most four colors (1853).
- Guthrie's conjecture was fallaciously proved by Alfred Kempe, a barrister, in 1878.
- In 1890, Percy Heawood (a mathematician) found an irreparable flaw in Kempe's proof.
- The Four Color Conjecture was solved in the affirmative in 1976 by Appel, Haken, and Koch by way of joining together theory and computation; the latter took hundreds of hours of computation time.
- The problem was reduced to one of checking a finite number of graphs (about 2000) for special properties.

The Four Color Theorem

- After the dust had settled, Appel, Haken, and Koch had proved

The Four Color Theorem

- After the dust had settled, Appel, Haken, and Koch had proved
- **Theorem.** Let G be any simple planar graph. Then $\chi(G) \leq 4$ and the bound is sharp.

The Four Color Theorem

- After the dust had settled, Appel, Haken, and Koch had proved
- **Theorem.** Let G be any simple planar graph. Then $\chi(G) \leq 4$ and the bound is sharp.
- **More Current News.** The proof of the Four Color Theorem was greatly reduced by Robertson, Seymour, and others in the mid 1990s.

The Four Color Theorem

- After the dust had settled, Appel, Haken, and Koch had proved
- **Theorem.** Let G be any simple planar graph. Then $\chi(G) \leq 4$ and the bound is sharp.
- **More Current News.** The proof of the Four Color Theorem was greatly reduced by Robertson, Seymour, and others in the mid 1990s.
- The number of configurations to check was reduced to about 500, and of course computers had become much faster in the intervening two decades.

The Four Color Theorem

- After the dust had settled, Appel, Haken, and Koch had proved
- **Theorem.** Let G be any simple planar graph. Then $\chi(G) \leq 4$ and the bound is sharp.
- **More Current News.** The proof of the Four Color Theorem was greatly reduced by Robertson, Seymour, and others in the mid 1990s.
- The number of configurations to check was reduced to about 500, and of course computers had become much faster in the intervening two decades.
- There is still no known computer-free proof of the Four Color Theorem.

Graph Theory and Graph Algorithms, continued

- We'll take a look at a planar graph coloring generalization due to Percy Heawood.

Graph Theory and Graph Algorithms, continued

- We'll take a look at a planar graph coloring generalization due to Percy Heawood.
- When Heawood found and could not repair the flaw in Kempe's four color argument,

Graph Theory and Graph Algorithms, continued

- We'll take a look at a planar graph coloring generalization due to Percy Heawood.

- When Heawood found and could not repair the flaw in Kempe's four color argument,

- he started thinking about and inventing graph color problems that he *could* solve.

Graph Theory and Graph Algorithms, continued

- We'll take a look at a planar graph coloring generalization due to Percy Heawood.
- When Heawood found and could not repair the flaw in Kempe's four color argument,
- he started thinking about and inventing graph color problems that he *could* solve.
- Here is one such problem.

A "New" Graph Coloring Problem (circa 1890)

- **Set-up.** Given a planar graph G, suppose each country consists of at most M united empires (called M-pires).

A "New" Graph Coloring Problem (circa 1890)

- **Set-up.** Given a planar graph G, suppose each country consists of at most M united empires (called M-pires).

- Thus an M-pire will consist of up to M independent (i.e., M mutually non-adjacent) vertices in G.

A “New” Graph Coloring Problem (circa 1890)

- **Set-up.** Given a planar graph G, suppose each country consists of at most M united empires (called M-pires).

- Thus an M-pire will consist of up to M independent (i.e., M mutually non-adjacent) vertices in G.

- The cartographers want to color all members of a single M-pire with one color,

A "New" Graph Coloring Problem (circa 1890)

- **Set-up.** Given a planar graph G, suppose each country consists of at most M united empires (called M-pires).
- Thus an M-pire will consist of up to M independent (i.e., M mutually non-adjacent) vertices in G.
- The cartographers want to color all members of a single M-pire with one color,
- but still want to color adjacent vertices with different colors.

A "New" Graph Coloring Problem (circa 1890)

- **Set-up.** Given a planar graph G, suppose each country consists of at most M united empires (called M-pires).
- Thus an M-pire will consist of up to M independent (i.e., M mutually non-adjacent) vertices in G.
- The cartographers want to color all members of a single M-pire with one color,
- but still want to color adjacent vertices with different colors.
- **Question.** What can be said about $\chi(G)$ when G is an M-pire graph?

What about $\chi(G)$ when G is an M-pire graph?

- **Theorem M.** An M-pire graph G can always be properly vertex-colored with at most $6M$ colors. That is, if G is an M-pire graph, then $\chi(G) \leq 6m$.

What about $\chi(G)$ when G is an M-pire graph?

- **Theorem M.** An M-pire graph G can always be properly vertex-colored with at most $6M$ colors. That is, if G is an M-pire graph, then $\chi(G) \leq 6m$.
- **Proof.** Recall that G is a planar graph, and let G^* be derived from G by
  1. making a single vertex for each of the at most M members of a single M-pire, and
  2. maintaining the original adjacencies, but removing multiple edges.

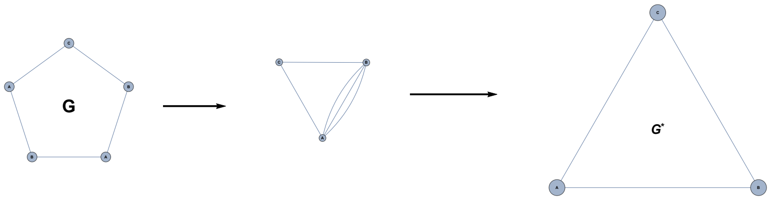
What about $\chi(G)$ when G is an M-pire graph?

- **Theorem M.** An M-pire graph G can always be properly vertex-colored with at most $6M$ colors. That is, if G is an M-pire graph, then $\chi(G) \leq 6m$.

- **Proof.** Recall that G is a planar graph, and let G^* be derived from G by
  1. making a single vertex for each of the at most M members of a single M-pire, and
  2. maintaining the original adjacencies, but removing multiple edges.
  3. There won't be any loops in G^*. Why?

Example of deriving G^* from G

Below G is an example of a 3-pire map; the names of the 3-pires are A, B, and C.



Back to $\chi(G)$ for G an M-pire graph

- **Claim.** $\chi(G^*) \leq 6M$ (which means $\chi(G) \leq M$).

Back to $\chi(G)$ for G an M-pire graph

- **Claim.** $\chi(G^*) \leq 6M$ (which means $\chi(G) \leq M$).

- **Notation.** Let $v^* = |V(G^*)|$, $e^* = |E(G^*)|$, $v = |V(G)|$, and $e = |E(G)|$.

Back to $\chi(G)$ for G an M-pire graph

- **Claim.** $\chi(G^*) \leq 6M$ (which means $\chi(G) \leq M$).
- **Notation.** Let $v^* = |V(G^*)|$, $e^* = |E(G^*)|$, $v = |V(G)|$, and $e = |E(G)|$.
- **Facts.**

Back to $\chi(G)$ for G an M-pire graph

- **Claim.** $\chi(G^*) \leq 6M$ (which means $\chi(G) \leq M$).

- **Notation.** Let $v^* = |V(G^*)|$, $e^* = |E(G^*)|$, $v = |V(G)|$, and $e = |E(G)|$.

- **Facts.**
  1. $Mv^* \geq v$ because each vertex in G^* corresponds to at most M vertices in G.

Back to $\chi(G)$ for G an M-pire graph

- **Claim.** $\chi(G^*) \leq 6M$ (which means $\chi(G) \leq M$).

- **Notation.** Let $v^* = |V(G^*)|$, $e^* = |E(G^*)|$, $v = |V(G)|$, and $e = |E(G)|$.

- **Facts.**
  1. $Mv^* \geq v$ because each vertex in G^* corresponds to at most M vertices in G.
  2. $e^* \leq e$ since we may have deleted some multiple edges.

Back to $\chi(G)$ for G an M-pire graph

- **Claim.** $\chi(G^*) \leq 6M$ (which means $\chi(G) \leq M$).

- **Notation.** Let $v^* = |V(G^*)|$, $e^* = |E(G^*)|$, $v = |V(G)|$, and $e = |E(G)|$.

- **Facts.**
  1. $Mv^* \geq v$ because each vertex in G^* corresponds to at most M vertices in G.
  2. $e^* \leq e$ since we may have deleted some multiple edges.

- **More Notation.** Let $n_1^*, n_2^*, \ldots, n_{v^*}^*$ be the vertices of G^*.

Back to $\chi(G)$ for G an M-pire graph

- **Claim.** $\chi(G^*) \leq 6M$ (which means $\chi(G) \leq M$).

- **Notation.** Let $v^* = |V(G^*)|$, $e^* = |E(G^*)|$, $v = |V(G)|$, and $e = |E(G)|$.

- **Facts.**
  1. $Mv^* \geq v$ because each vertex in G^* corresponds to at most M vertices in G.
  2. $e^* \leq e$ since we may have deleted some multiple edges.

- **More Notation.** Let $n_1^*, n_2^*, \ldots, n_{v^*}^*$ be the vertices of G^*.

- Then $deg(n_1^*) + deg(n_2^*) + \cdots + deg(n_{v^*}^*) = 2e^*$

$\chi(G)$ for G an M-pire graph, continued

- So far we have $deg(n_1^*) + deg(n_2^*) + \cdots + deg(n_{v^*}^*) = 2e^*$

$\chi(G)$ for G an M-pire graph, continued

- So far we have $deg(n_1^*) + deg(n_2^*) + \cdots + deg(n_{v^*}^*) = 2e^*$
- $\leq 2e$ (by Fact 2)

$\chi(G)$ for G an M-pire graph, continued

- So far we have $deg(n_1^*) + deg(n_2^*) + \cdots + deg(n_{v^*}^*) = 2e^*$
- $\leq 2e$ (by Fact 2)
- $\leq 6v - 12$ (follows from the proof of Lemma A)

$\chi(G)$ for G an M-pire graph, continued

- So far we have $deg(n_1^*) + deg(n_2^*) + \cdots + deg(n_{v^*}^*) = 2e^*$
- $\leq 2e$ (by Fact 2)
- $\leq 6v - 12$ (follows from the proof of Lemma A)
- $\leq 6Mv^* - 12$ (by Fact 1).

$\chi(G)$ for G an M-pire graph, continued

- So far we have $deg(n_1^*) + deg(n_2^*) + \cdots + deg(n_{v^*}^*) = 2e^*$
- $\leq 2e$ (by Fact 2)
- $\leq 6v - 12$ (follows from the proof of Lemma A)
- $\leq 6Mv^* - 12$ (by Fact 1).
- In that case, the average vertex degree in graph G^* is

$$\frac{\sum_{i=1}^{v^*} n_i^*}{v^*} = \frac{2e^*}{v^*} \leq \frac{2e}{v^*} \leq \frac{6v - 12}{v^*} \leq \frac{6Mv^*}{v^*} - \frac{12}{v^*}$$

$$= 6M - \frac{12}{v^*}.$$

$\chi(G)$ for G an M-pire graph, continued

- So far, the average vertex degree in G^* is at most $6M - \frac{12}{v^*}$

$\chi(G)$ for G an M-pire graph, continued

- So far, the average vertex degree in G^* is at most $6M - \frac{12}{v^*}$

- $\Rightarrow \exists$ a vertex of degree $6M - 1$ or less.

$\chi(G)$ for G an M-pire graph, continued

- So far, the average vertex degree in G^* is at most $6M - \frac{12}{v^*}$
- $\Rightarrow \exists$ a vertex of degree $6M - 1$ or less.
- Now proceed inductively in the same manner as the 6-color theorem (last lecture) to produce a proper $6M$-coloring of G^* and hence of G.

$\chi(G)$ for G an M-pire graph, continued

- So far, the average vertex degree in G^* is at most $6M - \frac{12}{v^*}$
- $\Rightarrow \exists$ a vertex of degree $6M - 1$ or less.
- Now proceed inductively in the same manner as the 6-color theorem (last lecture) to produce a proper $6M$-coloring of G^* and hence of G.
- Thus if G is an M-pire graph, then $\chi(G) \leq 6M$.

$\chi(G)$ for G an M-pire graph, continued

- So far, the average vertex degree in G^* is at most $6M - \frac{12}{v^*}$
- $\Rightarrow \exists$ a vertex of degree $6M - 1$ or less.
- Now proceed inductively in the same manner as the 6-color theorem (last lecture) to produce a proper $6M$-coloring of G^* and hence of G.
- Thus if G is an M-pire graph, then $\chi(G) \leq 6M$.
- We know that the bound is not best possible when $M = 1$ (why?), but it is best possible for all $M \geq 2$ as is shown by the next theorem.

Theorem M give best result for all $M \geq 2$

- **Theorem M' (G. RIngel and B. Jackson 1983).** For every $M > 1$ there is an M-pire graph whose chromatic number is exactly $6M$.

Theorem M give best result for all $M \geq 2$

- **Theorem M' (G. RIngel and B. Jackson 1983).** For every $M > 1$ there is an M-pire graph whose chromatic number is exactly $6M$.

- See the article *Variations on Ringel's Earth-Moon problem* by B. Jackson and G. Ringel, 1983:

Theorem M give best result for all $M \geq 2$

- **Theorem M' (G. RIngel and B. Jackson 1983).** For every $M > 1$ there is an M-pire graph whose chromatic number is exactly $6M$.
- See the article *Variations on Ringel's Earth-Moon problem* by B. Jackson and G. Ringel, 1983:
- `http://www.sciencedirect.com/science/article/pii/S0012365X99002782`

Theorem M give best result for all $M \geq 2$

- **Theorem M' (G. RIngel and B. Jackson 1983).** For every $M > 1$ there is an M-pire graph whose chromatic number is exactly $6M$.
- See the article *Variations on Ringel's Earth-Moon problem* by B. Jackson and G. Ringel, 1983:
- `http://www.sciencedirect.com/science/article/pii/S0012365X99002782`
- What exactly *is* Ringel's Earth-Moon question?

Ringel's Famous Earth-Moon Graph Coloring Question

- Earth wants to colonize its moon and cartographers wish to color the Earth-Moon map so that

Ringel's Famous Earth-Moon Graph Coloring Question

- Earth wants to colonize its moon and cartographers wish to color the Earth-Moon map so that

  1. adjacent countries on Earth are colored with distinct colors,

Ringel's Famous Earth-Moon Graph Coloring Question

- Earth wants to colonize its moon and cartographers wish to color the Earth-Moon map so that

  1. adjacent countries on Earth are colored with distinct colors,
  2. adjacent colonies on the moon are colored with distinct colors, and

Ringel's Famous Earth-Moon Graph Coloring Question

- Earth wants to colonize its moon and cartographers wish to color the Earth-Moon map so that

  1. adjacent countries on Earth are colored with distinct colors,
  2. adjacent colonies on the moon are colored with distinct colors, and
  3. a colony on the moon receives the same color as its corresponding country on Earth.

Ringel's Famous Earth-Moon Graph Coloring Question

- Earth wants to colonize its moon and cartographers wish to color the Earth-Moon map so that

  1. adjacent countries on Earth are colored with distinct colors,
  2. adjacent colonies on the moon are colored with distinct colors, and
  3. a colony on the moon receives the same color as its corresponding country on Earth.

- Disregarding the coloring for the moment, a graph so constructed can be decomposed into at most two planar graphs, each of which uses the same set of vertices.

Ringel's Famous Earth-Moon Graph Coloring Question

- Earth wants to colonize its moon and cartographers wish to color the Earth-Moon map so that

  1. adjacent countries on Earth are colored with distinct colors,
  2. adjacent colonies on the moon are colored with distinct colors, and
  3. a colony on the moon receives the same color as its corresponding country on Earth.

- Disregarding the coloring for the moment, a graph so constructed can be decomposed into at most two planar graphs, each of which uses the same set of vertices.

- **Definition.** If such an Earth-Moon graph G can be decomposed into two, and no fewer, planar graphs, then G is said to have thickness two.

Generalizing thickness

- **Definition T.** Let G be an undirected graph such that the edges of G can be partitioned into t sets, each of which induces a planar graph, and for which t is smallest possible. Then G is said to have thickness t, denoted by $\theta(G) = t$.

Generalizing thickness

- **Definition T.** Let G be an undirected graph such that the edges of G can be partitioned into t sets, each of which induces a planar graph, and for which t is smallest possible. Then G is said to have thickness t, denoted by $\theta(G) = t$.

- The notion of the thickness of a graph helps to measure the *near planarity* of a graph, and is a tool used in VLSI design, for example.

Thickness exercises

- $\theta(K_4) = 1$ (K_4 is the complete graph on four vertices).

Thickness exercises

- $\theta(K_4) = 1$ (K_4 is the complete graph on four vertices).
- If G is planar, what is $\theta(G)$?
- What is $\theta(K_5)$?

Thickness exercises

- $\theta(K_4) = 1$ (K_4 is the complete graph on four vertices).
- If G is planar, what is $\theta(G)$?
- What is $\theta(K_5)$?
- ... what is $\theta(K_8)$, $\theta(K_9)$?

Thickness exercises

- $\theta(K_4) = 1$ (K_4 is the complete graph on four vertices).
- If G is planar, what is $\theta(G)$?
- What is $\theta(K_5)$?
- ... what is $\theta(K_8)$, $\theta(K_9)$?
- What is $\theta(P)$, where P is the Petersen graph?

Open Research Problem

- A BIG open question is the following: what can be said about $\chi(G)$ when G is a thickness-2 graph?

Open Research Problem

- A BIG open question is the following: what can be said about $\chi(G)$ when G is a thickness-2 graph?
- Here is state-of-the-art knowledge:

Open Research Problem

- A BIG open question is the following: what can be said about $\chi(G)$ when G is a thickness-2 graph?
- Here is state-of-the-art knowledge:
- **Theorem E-M.** Let G be a thickness-2 graph. Then the largest possible chromatic number $\chi(G)$ over all possible thickness-2 graphs is one of 9,10,11, or 12.

Open Research Problem

- A BIG open question is the following: what can be said about $\chi(G)$ when G is a thickness-2 graph?
- Here is state-of-the-art knowledge:
- **Theorem E-M.** Let G be a thickness-2 graph. Then the largest possible chromatic number $\chi(G)$ over all possible thickness-2 graphs is one of 9,10,11, or 12.
- Can anybody reduce the list at all?

Proof of Theorem E-M

- A thickness-2 graph G is a 2-pire graph (why?),

Proof of Theorem E-M

- A thickness-2 graph G is a 2-pire graph (why?),
- which means $\chi(G) \leq 12$, and

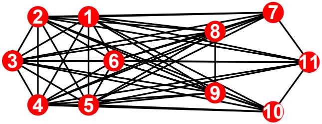
Proof of Theorem E-M

- A thickness-2 graph G is a 2-pire graph (why?),
- which means $\chi(G) \leq 12$, and
- Thom Sulanke found an Earth-Moon graph whose chromatic number is 9.

Proof of Theorem E-M

- A thickness-2 graph G is a 2-pire graph (why?),
- which means $\chi(G) \leq 12$, and
- Thom Sulanke found an Earth-Moon graph whose chromatic number is 9.
- **QED**

Sulanke's 9-chromatic thickness-2 graph



-

Figure: K_6 joined with C_5

Sulanke's 9-chromatic thickness-2 graph

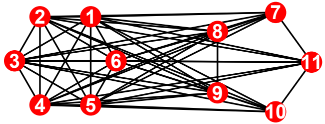
- 

Figure: K_6 joined with C_5

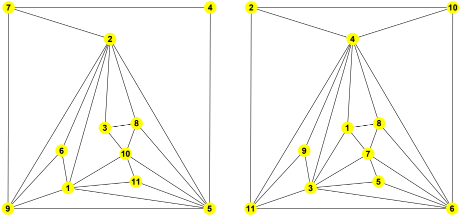
- 

Figure: thickness-2 decomposition

Thickness-2 application to chip testing

- **The Setting.** Test a printed circuit board (PCB) for certain kinds of errors.

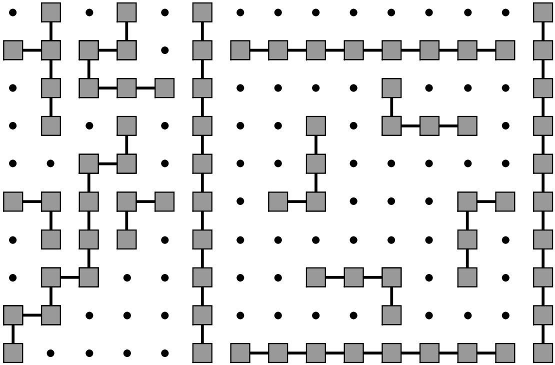
Thickness-2 application to chip testing

- **The Setting.** Test a printed circuit board (PCB) for certain kinds of errors.
- This problem was posed by Garey, Johnson, and So of AT&T labs in 1976.

Example of a PCB

- The vertices of the PCB are subsets of points on a grid, each of which is connected by a network of horizontal and vertical wires.

Example of a PCB

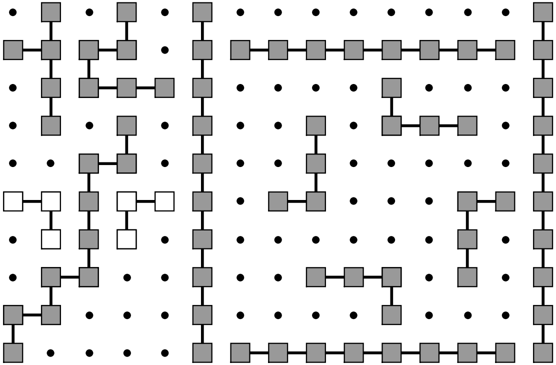
- The vertices of the PCB are subsets of points on a grid, each of which is connected by a network of horizontal and vertical wires.
- 

Example of a PCB error

- The white nodes of the PCB may be connected accidentally if both are connected to the node in between.

Example of a PCB error

- The white nodes of the PCB may be connected accidentally if both are connected to the node in between.



A real-life aside

- In September of 1989 the New York Times reported a PCB error in a \$12 PCB that caused a manufacturing delay in 2500 Air Force rockets, each worth (at that time) \$1,000,000.

A real-life aside

- In September of 1989 the New York Times reported a PCB error in a $12 PCB that caused a manufacturing delay in 2500 Air Force rockets, each worth (at that time) $1,000,000.

- For a copy of the NYT article, see the student webpage.

A real-life aside

- In September of 1989 the New York Times reported a PCB error in a $12 PCB that caused a manufacturing delay in 2500 Air Force rockets, each worth (at that time) $1,000,000.
- For a copy of the NYT article, see the student webpage.
- **Question.** How can we efficiently check for these types of errors?

Ideas for PCB testing

- **Idea 1.** Suppose our PCB has n nodes (i.e., n connected components). Then check every pair for accidental connections, which will take time $O(n^2)$.

Ideas for PCB testing

- **Idea 1.** Suppose our PCB has n nodes (i.e., n connected components). Then check every pair for accidental connections, which will take time $O(n^2)$.

- **Idea 2.** Create the visibility graph of the PCB and call it G.

Ideas for PCB testing

- **Idea 1.** Suppose our PCB has n nodes (i.e., n connected components). Then check every pair for accidental connections, which will take time $O(n^2)$.

- **Idea 2.** Create the visibility graph of the PCB and call it G.

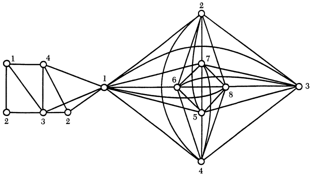
- 

Figure: 8-colored visibility graph of G

Ideas for PCB testing

- **Idea 1.** Suppose our PCB has n nodes (i.e., n connected components). Then check every pair for accidental connections, which will take time $O(n^2)$.

- **Idea 2.** Create the visibility graph of the PCB and call it G.

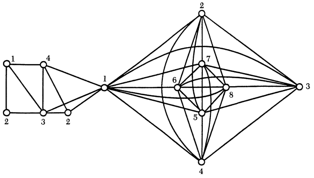
- 

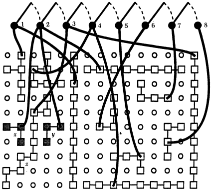
Figure: 8-colored visibility graph of G

- In our example, $\chi(G) = 8$ (why?) and in general we can properly color G with 12 or fewer colors (why?).

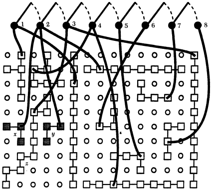
Idea 2 for PCB testing

- Create a “supernet” with one giant “claw” for each color of G. A claw connects to each node of the PCB of the same color.

Idea 2 for PCB testing

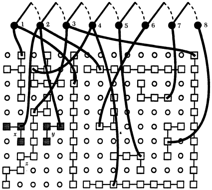
- Create a “supernet” with one giant “claw” for each color of G. A claw connects to each node of the PCB of the same color.
- 

Idea 2 for PCB testing

- Create a "supernet" with one giant "claw" for each color of G. A claw connects to each node of the PCB of the same color.
- 

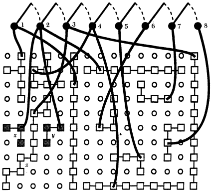
- Engage Net 1. If a circuit is completed then there is an error.

Idea 2 for PCB testing

- Create a "supernet" with one giant "claw" for each color of G. A claw connects to each node of the PCB of the same color.
- 

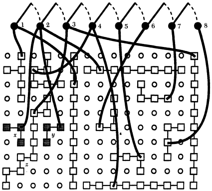
- Engage Net 1. If a circuit is completed then there is an error.
- Otherwise engage Net 1 and Net 2. If a circuit is completed then there is an error.

Idea 2 for PCB testing

- Create a "supernet" with one giant "claw" for each color of G. A claw connects to each node of the PCB of the same color.
- 

- Engage Net 1. If a circuit is completed then there is an error.
- Otherwise engage Net 1 and Net 2. If a circuit is completed then there is an error.
- ⋮

Idea 2 for PCB testing

- Create a "supernet" with one giant "claw" for each color of G. A claw connects to each node of the PCB of the same color.
  - 

- Engage Net 1. If a circuit is completed then there is an error.
- Otherwise engage Net 1 and Net 2. If a circuit is completed then there is an error.
- $\vdots$
- Engage all possible pairs of Nets 1-8. There are $\frac{8\times 7}{2} = 28$ possibilities.

Ideas 2 and 3 for PCB testing

- In general, if you've used 12 colors, there will be 66 checks, which has reduced the run time from $O(n^2)$ to $\theta(1)$.

Ideas 2 and 3 for PCB testing

- In general, if you've used 12 colors, there will be 66 checks, which has reduced the run time from $O(n^2)$ to $\theta(1)$.

- **Idea 3.** Use the same supernet.

Ideas 2 and 3 for PCB testing

- In general, if you've used 12 colors, there will be 66 checks, which has reduced the run time from $O(n^2)$ to $\theta(1)$.

- **Idea 3.** Use the same supernet.
- Connect Net 1. If a circuit is completed, then thee is an error.

Ideas 2 and 3 for PCB testing

- In general, if you've used 12 colors, there will be 66 checks, which has reduced the run time from $O(n^2)$ to $\theta(1)$.

- **Idea 3.** Use the same supernet.
- Connect Net 1. If a circuit is completed, then thee is an error.
- Otherwise connect Net 1 and Net 2 to form Net $\overline{12}$.

Ideas 2 and 3 for PCB testing

- In general, if you've used 12 colors, there will be 66 checks, which has reduced the run time from $O(n^2)$ to $\theta(1)$.

- **Idea 3.** Use the same supernet.
- Connect Net 1. If a circuit is completed, then thee is an error.
- Otherwise connect Net 1 and Net 2 to form Net $\overline{12}$.
- Connect Net $\overline{12}$ to the PCB; if a circuit is completed, there is an error.

Ideas 2 and 3 for PCB testing

- In general, if you've used 12 colors, there will be 66 checks, which has reduced the run time from $O(n^2)$ to $\theta(1)$.
- **Idea 3.** Use the same supernet.
- Connect Net 1. If a circuit is completed, then thee is an error.
- Otherwise connect Net 1 and Net 2 to form Net $\overline{12}$.
- Connect Net $\overline{12}$ to the PCB; if a circuit is completed, there is an error.
- Keep going until you have made Net $\overline{123\ldots12}$.

Ideas 2 and 3 for PCB testing

- In general, if you've used 12 colors, there will be 66 checks, which has reduced the run time from $O(n^2)$ to $\theta(1)$.
- **Idea 3.** Use the same supernet.
- Connect Net 1. If a circuit is completed, then thee is an error.
- Otherwise connect Net 1 and Net 2 to form Net $\overline{12}$.
- Connect Net $\overline{12}$ to the PCB; if a circuit is completed, there is an error.
- Keep going until you have made Net $\overline{123\ldots12}$.
- Total number of checks: 11. Whew.

See the student webpage for a copy of the journal article that contains the full treatment of the M-pire problem, *Coloring Ordinary Maps, Maps of Empires, and Maps of the Moon*.

Next Topic: Graph Algorithms

Single Source Shortest Path Algorithms

A Scenario

- Find a good route from home to a restaurant in a city.

A Scenario

- Find a good route from home to a restaurant in a city.

- The street-intersections are vertices and the edges are street segments (like blocks).

A Scenario

- Find a good route from home to a restaurant in a city.
- The street-intersections are vertices and the edges are street segments (like blocks).
- We will talk about undirected graphs, but there are many variations (one-way streets, for example).

A Scenario

- Find a good route from home to a restaurant in a city.
- The street-intersections are vertices and the edges are street segments (like blocks).
- We will talk about undirected graphs, but there are many variations (one-way streets, for example).
- The edges will be weighted with a cost (time delay or distance, for example).

A Scenario

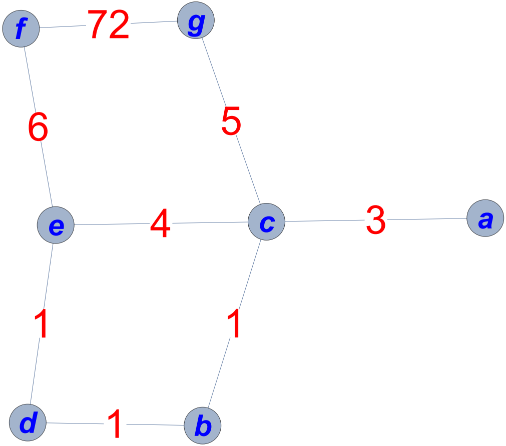
- Find a good route from home to a restaurant in a city.
- The street-intersections are vertices and the edges are street segments (like blocks).
- We will talk about undirected graphs, but there are many variations (one-way streets, for example).
- The edges will be weighted with a cost (time delay or distance, for example).
- **Problem.** Find the least costly path from home to the restaurant.

Example

- Find the least costly path from vertex a to vertex f below.

Example

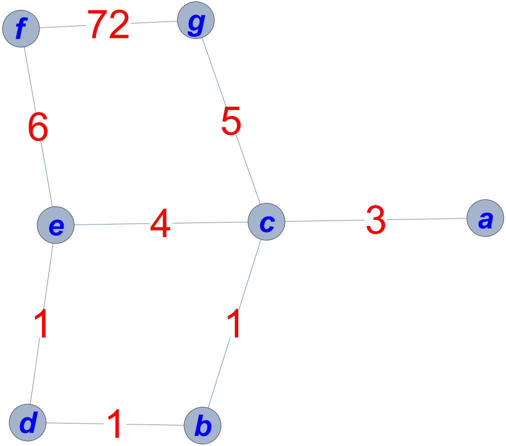
- Find the least costly path from vertex a to vertex f below.



-

Example

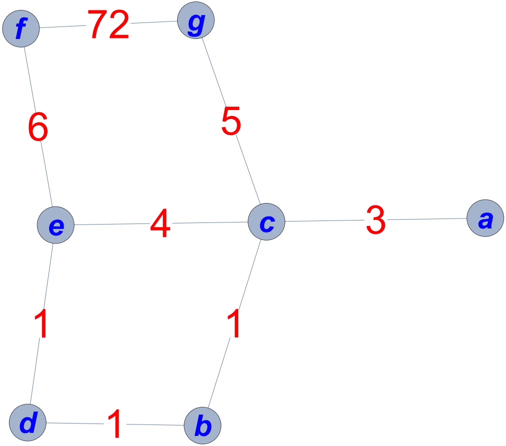
- Find the least costly path from vertex a to vertex f below.



-
- **Answer.** $a \to c \to b \to d \to e \to f$.

Example

- Find the least costly path from vertex a to vertex f below.



- **Answer.** $a \to c \to b \to d \to e \to f$.
- **Total Weight** $= 12$.

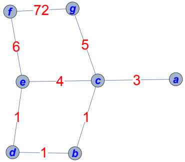
Informal Definition for Weight of an Edge

- The **edge weight function** ω of a graph G is a function from the edge set $E(G)$ to the set containing all possible edge weights.

Informal Definition for Weight of an Edge

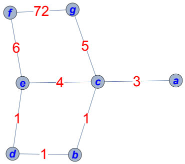
- The **edge weight function** ω of a graph G is a function from the edge set $E(G)$ to the set containing all possible edge weights.
- For example $\omega : E(G) \rightarrow \mathbb{R} \cup \infty$ is a common scenario.

Informal Definition for Weight of an Edge

- The **edge weight function** ω of a graph G is a function from the edge set $E(G)$ to the set containing all possible edge weights.
- For example $\omega : E(G) \to \mathbb{R} \cup \infty$ is a common scenario.
- 

- In our example, we have $\omega(ac) = 3$, $\omega(bd) = 1$, $\omega(gf) = 72$, etc.

Informal Definition for Weight of an Edge

- The **edge weight function** ω of a graph G is a function from the edge set $E(G)$ to the set containing all possible edge weights.
- For example $\omega : E(G) \to \mathbb{R} \cup \infty$ is a common scenario.
- 

- In our example, we have $\omega(ac) = 3$, $\omega(bd) = 1$, $\omega(gf) = 72$, etc.
- Importantly, if $xy \notin E(G)$ then we define $\omega(xy) = \infty$ so that there is no chance of a non-edge being chosen in a shortest path algorithm.

Single Source Shortest Path

- **Problem.** We want to find all possible shortest paths from a single vertex (home) to all other vertices in the graph (restaurants).

Single Source Shortest Path

- **Problem.** We want to find all possible shortest paths from a single vertex (home) to all other vertices in the graph (restaurants).

- **Notation.** To that end, given a weighted graph G with with $v_i, v_j \in V(G)$, define the **distance** from v_i to v_j, denoted $d(v_i, v_j)$, to be the weight of the least costly path from v_i to v_j.

Single Source Shortest Path

- **Problem.** We want to find all possible shortest paths from a single vertex (home) to all other vertices in the graph (restaurants).

- **Notation.** To that end, given a weighted graph G with with $v_i, v_j \in V(G)$, define the **distance** from v_i to v_j, denoted $d(v_i, v_j)$, to be the weight of the least costly path from v_i to v_j.

- If there is no path from v_i to v_j define $d(v_i, v_j) = \infty$.

A tool: spanning tree

- **Definition.** Let G be a simple, connected, undirected graph with n vertices. Then a **spanning tree**, T, of G is any connected acyclic subgraph of G with n vertices.

A tool: spanning tree

- **Definition.** Let G be a simple, connected, undirected graph with n vertices. Then a **spanning tree**, T, of G is any connected acyclic subgraph of G with n vertices.

- **Theorem.** Let G be an undirected graph with n vertices. Then T is a spanning tree of G if and only if T is a connected subgraph of G with $n - 1$ edges.

A tool: spanning tree

- **Definition.** Let G be a simple, connected, undirected graph with n vertices. Then a **spanning tree**, T, of G is any connected acyclic subgraph of G with n vertices.
- **Theorem.** Let G be an undirected graph with n vertices. Then T is a spanning tree of G if and only if T is a connected subgraph of G with $n - 1$ edges.
- For a proof of the above theorem, take CSCI 5408 (Applied Graph Theory) in the spring.

Next: A Tree Growing Algorithm