Graph Theory Midterm Review

Practice Exam Solutions

Practice Exam Problem 1

Problem. Consider the following in-degree and out-degree sequences for some hypothetical directed graph *D*. These sequences are not in any particular vertex order, so $S^+(1)$ and $S^-(1)$ don't necessarily refer to the same vertex. Is an Eulerian circuit possible on *D*? Justify your response. $S^+ = \{2, 4, 4, 2, 2, 6, 2, 8, 3\}$, $S^- = \{3, 2, 6, 2, 2, 4, 4, 2, 8\}$

Theorems and Definitions.

- Eulerian circuits are a closed trail which start and end at the same vertex and contain all edges of a graph
- For *undirected* graphs, a graph can only be Eulerian if it has even degree

Practice Exam Problem 1

Solution. For *directed* graphs, we require a condition stronger than even (total) degree. A directed graph is Eulerian iff the in degree of all vertices is equal to their out degree. This is because:

- The starting vertex must be exited at least once and entered at least once (as the circuit starts and ends)
- All vertices must be entered and exited an equal number of times, else we could only have an Eulerian path at most

From the given degree sequences, it is possible to have a *D* with equal in and out degree for all vertices so an Eulerian circuit is possible.

Practice Exam Problem 2

Problem. Prove that connected G contains a cycle iff $|V(G)| \leq |E(G)|$.

Solution. We first prove if a connected graph has $|V(G)| \leq |E(G)|$ then it has a cycle. This was similar to a problem in Homework 1. We prove by contradiction.

- G is connected, acyclic graph (tree) has $|E(G)| = |V(G)| - 1$ edges and is maximally acyclic
- We have at least $|V(G)|$ edges so one more edge means G has a cycle

Thus, we have if a connected graph has $|V(G)| \leq |E(G)|$ then it has a cycle.

Practice Exam Problem 2

Now we prove if a connected G has a cycle then it has $|V(G)| \leq |E(G)|$ via strong induction on $|E(G)|$.

Basis: $P(1)$ is a trivial graph with a self-loop. We have $|V(G)|=1$ and $|E(G)|=1$ so clearly $|V(G)| \leq |E(G)|$

I.H.: All connected graphs with $k=|E(G)|$ and $k < n$ and a cycle have $|V(G)| \leq |E(G)|$.

Let us have a graph $P(n)$ with a cycle. We take $P(k)=P(n) \cdot e$. Edge contraction will retain cycles and can't disconnect $P(k)$ so by the IH we have $|V(P(k))| \leq |E(P(k))|$. By re-expand the edge e, we get $|V(P(n))|=|V(P(k))|+1$ and $|E(P(n))|=|E(P(k))|+1$. Thus, we have $|V(P(n))| \leq |E(P(n))|$.

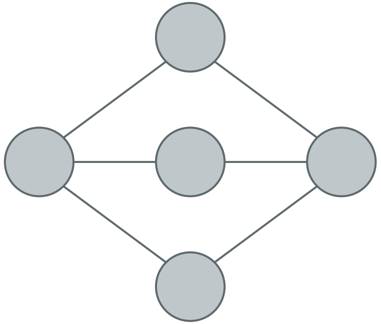
Now that we've proven both direction we have that a connected G contains a cycle iff $|V(G)| \leq |E(G)|$.

Practice Exam Problem 3

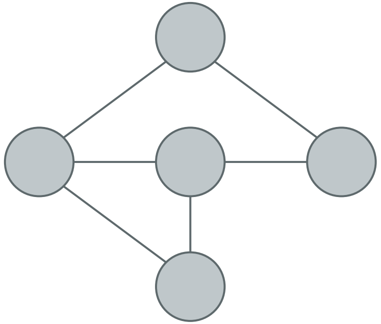
Problem. Prove or disprove: Havel-Hakimi can generate all possible graph configurations.

Solution. During each iteration of Havel-Hakimi, we always connect the largest degree vertex in the sequence, to the next largest degree vertices. So we can *disprove* that Havel-Hakimi can generate all possible graph configurations by providing a counterexample where the largest degree vertices are not connected.

S = {3 3 2 2 2}
Counterexample:



Havel-Hakimi generates:

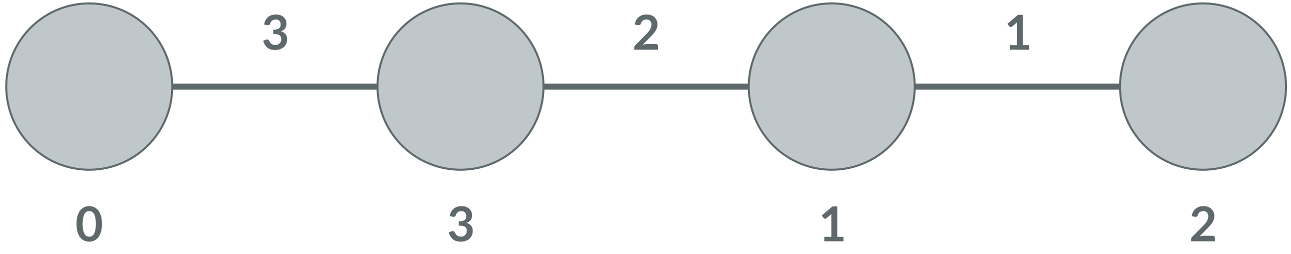


Practice Exam Problem 4

Problem. Draw and gracefully label a connected graph of at least 4 vertices.

Things to Remember.

- Both vertex and edge labels must be *unique* for graceful labeling
- An edge label is the (absolute) difference of its vertices' labels



Practice Exam Problem 5

Problem. Consider the following enumerative questions for undirected graphs, where loopy graphs and multi-graphs are proper supersets of simple graphs and we're considering a vertex set of cardinality *n*:

(a) How many possible loopy graphs are there?

(b) How many possible loopy multigraphs, with a maximum number of multi-edges of 2?

Practice Exam Problem 5

Solution.

Note, we will count isomorphic graphs as well for this.

(a) We have n possible unique loops and n(n-1)/2 possible unique edges (n vertices in pairs of 2 can be paired n(n-1)/2 ways)

Every edge can either exist in a possibility or not. This is a *binary* choice.

So we have $2^{n(n-1)/2+n}$ possible loopy graphs.

Practice Exam Problem 5

Solution.

(b) We again have n possible unique loops and n(n-1)/2 possible unique edges

Now for every possible edge location we can either have 0, 1, or 2 edges.

So we have $3^{n(n-1)/2+n}$ possible loopy multi-graphs if we allow for multi-loops, otherwise loops are a binary choice still and we have $3^{n(n-1)/2}2^n$

Practice Exam Problem 6

Problem. G has a unique weight for each edge. Prove that G has a unique minimum spanning tree.

Solution. We have proven the correctness of Kruskal's minimum spanning tree algorithm.

- The algorithm uses edges in sorted order
- As edge weights are unique, this order will be strict so there is only one possible output
- Output is guaranteed to be a MST so only this unique output is possible

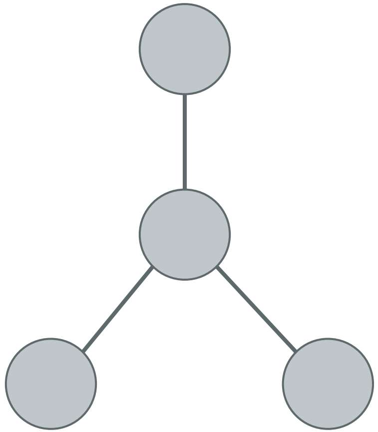
Thus, G has a minimum spanning tree.

This could also be done as a proof by contradiction.

Practice Exam Problem 7

Problem. Prove or disprove: Every tree with an even number of vertices has a perfect matching.

Solution. Disprove by counterexample. We can see that though |E(G)| is even, it fails Tutte's theorem so no perfect match.



Practice Exam Problem 8

Problem. Consider graph G, and prove that $\forall v \in G : G - v$ has a perfect match iff $|V(G)|$ is odd and $o(G - S) \le |S| : \forall S \subseteq V(G)$.

Solution. We first prove if $\forall v \in G : G - v$ has a perfect match then $|V(G)|$ is odd and $o(G - S) \le |S| : \forall S \subseteq V(G)$.

- A perfect match requires even number of vertices so $|V(G\text{-}v)|$ is even then $|V(G)|$ is odd
- Let $G'=G-v$. By Tutte's, we know $\forall S' \subseteq V(G')$, $o(G' - S') \le |S'|$.
- Let $S=S'+v$. We have $G - S = G' - S'$ so $o(G - S) = o(G' - S')$ and $|S'|=|S|-1$. Then $o(G - S)=o(G' - S')\le|S'|=|S|-1$ so $o(G - S)\le|S|$

Practice Exam Problem 8

Now we prove if $|V(G)|$ is odd and $o(G - S) \leq |S| : \forall S \subseteq V(G)$ then $\forall v \in G : G - v$ has a perfect match.

- Let $G' = G - v$ and $S = S' + v$ where $S' \subseteq V(G')$
- Since $|V(G)|$ is odd then $|S|$ and $o(G - S)$ have a different parity so we instead have $o(G - S) \leq |S| - 1$
- We have that $o(G' - S') = o(G - S) \leq |S| - 1 = |S'|$

Tutte's theorem then holds for G' so it has a perfect match.

Practice Exam Problem 9

Problem. Prove that every 3-regular graph without a cut edge has a perfect match.

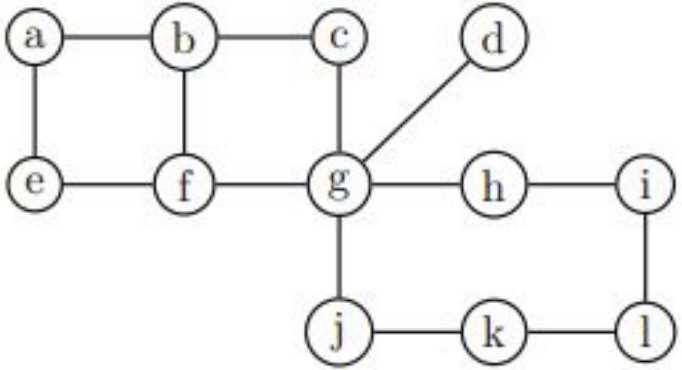
Solution. Consider 3-regular graph G without a cut edge. Consider some $S \subseteq V(G)$ and for G - S, let's label our odd components as $H_1, H_2, \ldots, H_k$.

- We can have at most $3|S|$ edges from S to all odd components
- We need at least 2 edges from each odd component to S as there is no single cut edge
- So the minimum number of edges from S to all odd components is $2o(G-S)$
- Odd components have an odd $|V(H_i)|$ by definition, the degree sum of an odd component is $3|V(H_i)|-2$ which is odd. The minimum cut must then be 3 edges
- So the minimum number of edges from S to all odd components is $3o(G-S)$
- $3o(G - S) \leq$ actual size of cut $\leq 3|S|$ so clearly $o(G - S) \leq |S|$ and we have a perfect match by Tutte's theorem

Practice Exam Problem 10

Problem. Consider graph G below. We'll consider a few questions about G.

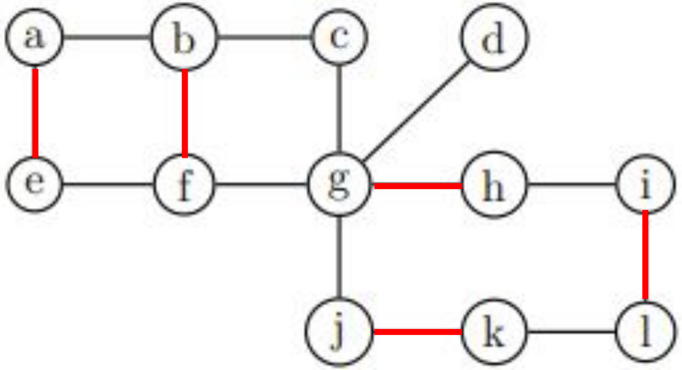
(a) Identify a maximum match on G. Prove that this match is optimal.
(b) Identify a minimum vertex cover on G. Prove that this vertex cover is optimal.
(c) Identify a minimum dominating set on G. Prove that this dominating set is optimal.



Practice Exam Problem 10

Solution.

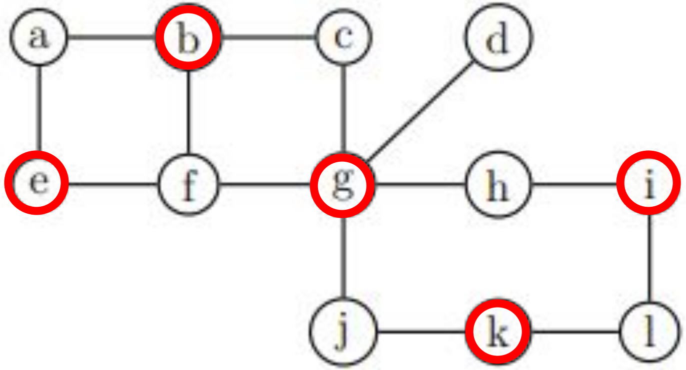
(a) Notice we have cut vertex g whose removal results in 3 odd connected components. By Tutte's we do not have a perfect match so our match is maximum.



Practice Exam Problem 10

Solution.

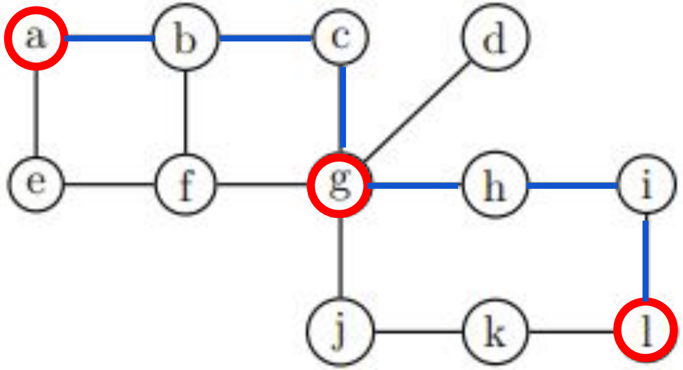
(a) Note, as G has no odd cycles then it is bipartite. By the König-Egerváry theorem, the size of a minimum vertex cover is the size of a maximum match.



Practice Exam Problem 10

Solution.

(a) To find the minimum dominating set, consider the longest shortest path (or one of). For G, it is a path P of length 6. If an included vertex occurs every third vertex, we need at least ceil(|V(P)|/3) = ceil(7/3) = 3 vertices to cover this path.



Homework 3 Solutions

HW 3 Problem 1

Problem. True or False: Tree T has at most one unique perfect matching. Prove your response.

Solution. Proof by contradiction. Assume T has 2 distinct perfect matches, M_1 and M_2, on T. Consider the symmetric difference $F=M_1 \Delta M_2$

- Leaves are always matched to sole neighbor so no edges of leaves are in F
- Take T'=T-Q where Q is the set of leaves and their sole neighbor
- T' has a perfect match in M_1 or M_2 and all leaves of T' will be matched to their neighbor so those edges again won't be in F
- We repeat until no edges remain and observe that F will be empty

M_1 and M_2 must be equal so any perfect match on a tree is unique.

HW3 Problem 1

Other Approaches. This problem can also be done via induction or by constructing the symmetric difference as V(F)=V(G) and E(F)=$M_1 \Delta M_2$. This allows us to consider the degree of each vertex in F. When a vertex is either saturated by the same edge in M_1 and M_2 and has degree 0 or it is saturated by different edges and has degree 2. By the connectivity of a tree, the degree 2 vertices must be connected and form a cycle which goes against the definition of a tree. Thus, there must be no degree 2 vertices so a tree has at most one unique perfect match.

HW 3 Problem 2

Problem. Consider maximum match M on G. Prove that every maximal match M’ has cardinality bounded by |M’| ≥ |M|/2 .

Solution. Assume instead we have |M’|<|M|/2.

- For any edge e=(u,v) in M’, there is at least 1 edge and at most 2 edges in M that are incident to u or v
- So at most 2|M’| edges in M are incident to a vertex in V(M’)
- We can rewrite our initial assumption as 2|M’|<|M|
- This implies 2|M’| vertices are saturated by an edge in M but the inequality implies there is at least one edge in M not incident to a vertex in M’
- This edge can be added to M’ which contradicts that M’ is maximal

Thus, we must have |M’| ≥ |M|/2 .

HW 3 Problem 3

Problem. Prove that the complement of any vertex cover on a simple undirected graph is an independent set.

Solution. We have vertex cover C and complement $\bar{C}$.

- Every edge in G is incident on at least one vertex v in C (at least one endpoint of every edge is in C)
- So every edge is incident on *at most* one vertex in $\bar{C}$
- By this bound we cannot have an edge between any two vertices of $\bar{C}$

Thus, $\bar{C}$ is an independent set.

This can also be done with a proof by contradiction.

HW 3 Problem 4

Problem. Consider graph G where $\forall v \in V(G) : d(v) = k$, $|V(G)|$ is even, and G remains connected after the deletion of any $(k-2)$ edges. Prove that G has a perfect match.

Solution. Recall that for Tutte's theorem we have $o(G - S) \leq |S| : \forall S \subseteq V(G)$.

- Consider some S where $|S| \geq 1$ and $G' = G - S$. Let H be an odd component of G'
- By our connectivity, H must be connected by at least $k - 1 \leq x$ edges
- The degree sum of H is $k|V(G)| - x = 2|E(H)|$ so k and x must both be even or odd
- By this parity, since we have $k - 1 \leq x$ then we also have $k \leq x$
- This holds for all odd components of G'
- Thus, we have $k*o(G - S)$ edges from S to all odd components
- The degree sum of S is $k|S|$, we have at most $k|S|$ edges to all odd components
- $k*o(G - S) \leq k|S|$ so $o(G - S) \leq |S|$ which holds for all S and k

Thus, G has a perfect match.

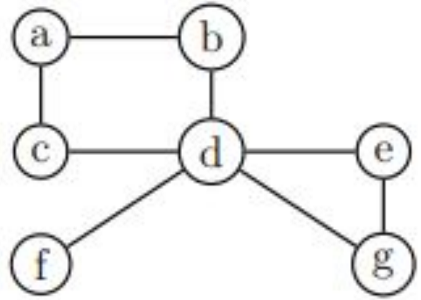
HW 3 Problem 5

Problem. Consider the below graph.

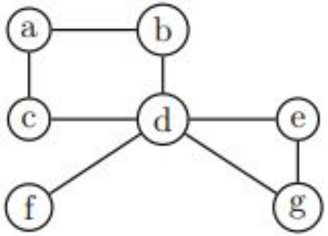
(a) Provide an edge cover F and vertex cover C for the below graph.

(b) Prove whether it possible to draw a perfect match M, such that F = M.

(c) Provide the complement of C, and show that it is an independent set.



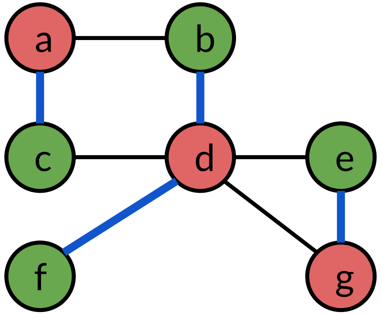
HW 3 Problem 5



(a) We could take our vertex cover to be C={a, d, g} and edge cover to be F={(a, c), (b, d), (d, f), (e, g)}.

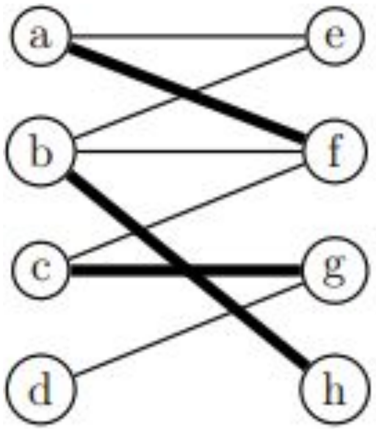
(b) It's not possible to draw a perfect match. Remember Tutte's theorem has o(G-S) for all S subsetting V(G). If we take S={d} then we have 2 odd components in G-S

(c) By the vertex cover from (a), we have complement $\bar{C}$={b, c, e, f}. As no edge exists between these vertices, it is clearly an independent set.



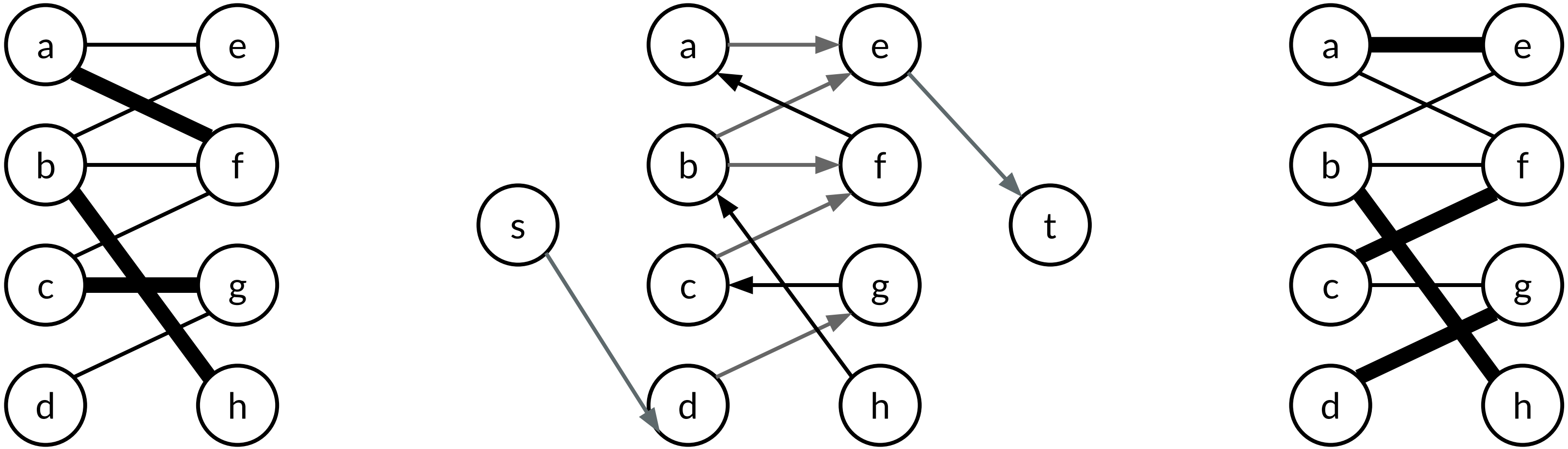
HW 3 Problem 6

Problem. Demonstrate a single iteration of our *M*-augmenting paths algorithm for the bipartite graph below to increase the size of the match *M* given in bold on the graph below. Explicitly show your steps



HW 3 Problem 6

Solution. With BFS we find shortest path s□d□g□c□f□a□e□t. We can swap our edges to get the following improved match.



HW 3 Problem 7

Problem. For each of the following values of k, construct a k-regular graph that does not have a perfect match. (**v1.1**) If that is not possible, prove why not.

(a) $k = 1$

(b) $k = 2$

(c) $k = 3$

HW 3 Problem 7

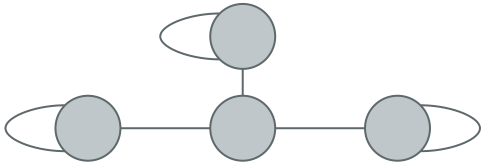
Solution.

(a) *k = 1*. The only 1-regular graph is made only of K_2 components which trivially always has a perfect match,

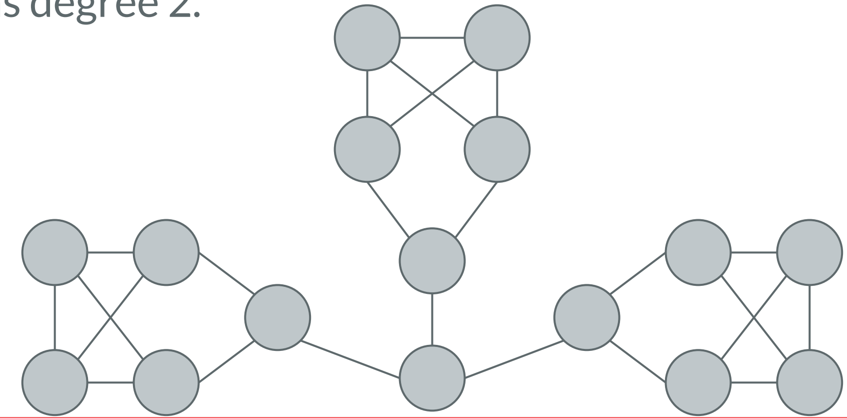
(b) *k = 2*. An odd cycle is 2-regular and doesn't have a match.

HW 3 Problem 7

(c) $k = 3$. As we don't require our counterexample to be simple, we could use the following:



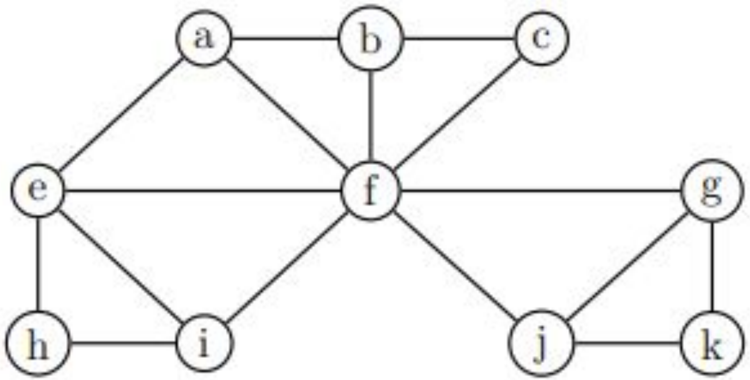
If we don't use simple, we could find an instance with a block-cutpoint graph using the above shape where BiCCs replace the loopy vertices. The BiCCs should be 3-regular for all but one vertex which has degree 2.



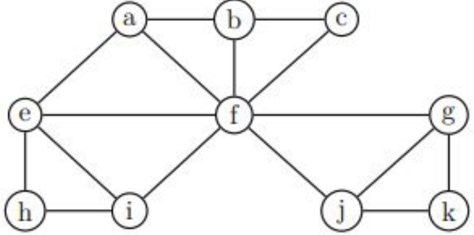
Weekly Problem 7 Solutions

WP 7 Problem 1

Problem. Is a closed ear decomposition of the below graph possible? What about an open ear decomposition? Draw one for each if possible. What does this prove about its connectivity and edge-connectivity?



WP 7 Problem 1



Solution. We could do the following closed ear decomposition:

P_0={(f, i), (i, h), (h, e), (e, a), (a, b), (b, c), (c, f)}

P_1=(e, i)

P_2=(e, f)

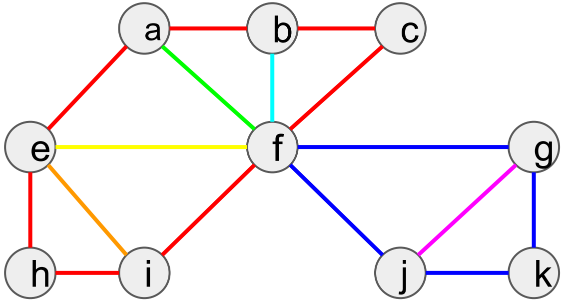
P_3=(f, a)

P_4=(f, b)

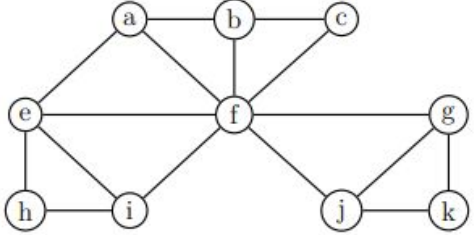
P_5={(f, j), (j, k), (k, g), (g, f)}

P_6=(j, g)

We can see our decomposition with colors on the right.



WP 7 Problem 1



An open ear decomposition is not possible

- Graph must be 2-connected to have an open ear decomposition
- Graph is 2-connected iff it is connected and has no cut vertex
- We see *f* is a cut vertex in *G*
- *G* is then not 2-connected and has no open ear decomposition

As it is connected but not 2-connected then we have connectivity $\kappa(G)=1$ and since we do have a closed ear decomposition then we have edge-connectivity $\kappa'(G)\geq 2$

WP 7 Problem 2

Problem. Graph G has the following properties:

(a) Maximum degree $\Delta(G) = 4$.

(b) Minimum degree $\delta(G) = 2$.

(c) $\forall u, v \in V(G) : \exists$ a u, v-path.

(d) $\forall u, v \in V(G) : \exists$ a closed u, v-trail.

Put tight upper and lower bounds on k, k' for which G could be k-connected and k'-edge-connected, given these properties. Prove your responses.

WP 7 Problem 2

Solution. What do we know from our properties?

(a) *Maximum degree* $\Delta(G) = 4$. This actually tells us nothing.

(b) *Minimum degree* $\delta(G) = 2$. We know that our minimum degree upper bounds the connectivity and edge-connectivity

(c) $\forall u, v \in V(G): \exists$ *a u, v-path*. *G* is connected but not necessarily 2-connected.

(d) $\forall u, v \in V(G): \exists$ *a closed u, v-trail*. *G* is 2-edge-connected.

So we have $k'=2$ and $2 \geq k \geq 1$.

[If (c) specified a *closed* path then we could say *G* is 2-connected.]

Midterm Tips and Reminders

Tips and Reminder

- A graph doesn't need to be connected to be simple, it just can't have loops or multi-edges
- When no assumptions are given about a graph, you must show the required property holds whether it is simple, connected, disconnected, etc
- When asked to prove if something is *possible*, you must provide an example where it is true or prove it can't be true for any case
- When proving a graph is a tree, you only need to prove it is connected and acyclic or prove that any other sufficient property is true
- Problems which use Tutte's theorem generally reference the degrees of vertices and whether a graph has a perfect match
- It may be useful to include common enumeration formulas on your crib sheet, such as the binomial coefficient