21/09/20 ABSENT
TSP and Chinese Postman Problem

23/09/20 ONLY doubts in class test I.

25/09/20 # Cut-Set

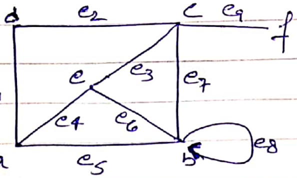
In a connected graph G, a cut-set, C, is a set of edges, whose removal leaves G disconnected; provided no proper subset of C leaves the graph G disconnected

$S = \{e_{i_1}, e_{i_2} \ldots \}$

$G(V, E) \rightarrow$ connected
$G(V, E \setminus S) \rightarrow$ disconnected
if $\nexists S' \subset S$ $G(V, E \setminus S')$ is disconnected
then S is cut-set.

☆ Minimal set of edges of graph (connected) G, whose removal leaves G disconnected

eg



G.

$S_1 : \{e_9\}$
$S_2 : \{e_1, e_2\}$
$S_3 : \{e_4, e_6, e_3\}$

various cut-sets for G

☆ Minimal set of edges in (connected) graph G whose removal reduces the rank by 1.

→ used to identify weak points
eg. the graph represents connections (roads / wires etc)
cut set ⇒ ease of disruption.

characterisation of cut-set

☆ Th — Every cut-set in a connected graph G must contain at least one branch of every spanning tree of G.

Proof — let S be a cut set of G, and ∃ a spanning tree T such that no branch of T is in S.

$\Rightarrow T \subseteq G \setminus S$ as S does not contain edges in T.

$\Rightarrow G \setminus S$ is connected $\Rightarrow$ contradiction

$\Rightarrow$ ~~S is~~ $\Rightarrow$ S must have at least one edge of T

$\Rightarrow$ Every cut set must contain at least one branch of every spanning tree

☆ Th — In a connected graph G, any minimal set of edges containing at least one branch of every spanning tree is a cut set.

Proof — let Q be a minimal set of edges containing at least one branch of every spanning tree of G.

$G \setminus Q$ is a disconnected graph ($\because \nexists$ any spanning tree)

for any $e \in Q$

$G \setminus Q + \{e\}$ contains at least one spanning tree as e belongs to at least one spanning tree

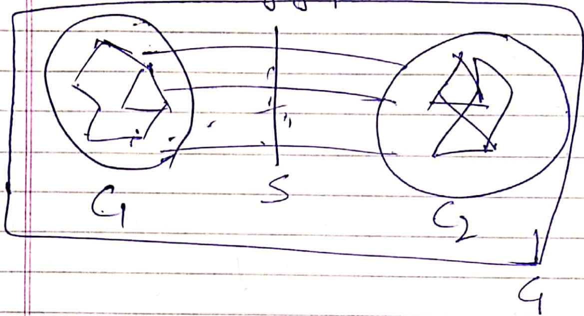
$\Rightarrow$ Q is minimal set of edges whose removal makes G disconnected

$\Rightarrow$ Q is a cut set.

Thm Every ~~tree~~ circuit has an even no of edges in common with any cut set.

$G <$ circuit Γ, cut set S $\quad |E(\Gamma \cap S)| = \text{even}$

let an arbitrary graph G



if circuit Γ in G_1 or G_2 only
$|E(\Gamma \cap S)| = 0 \Rightarrow$ even

if circuit Γ across G_1 & G_2
$|E(\Gamma \cap S)| = 2x \Rightarrow$ even $\quad x \in \mathbb{N} + \{0\}$
as Γ is closed, $|E(\Gamma \cap S)|$ are $2 *$ no. of crossovers ~~from $G_{1/2}$ to $G_{2/1}$ as Γ is closed.~~
~~if $\Gamma : v_i \in G_1 - v_{i_1} - v_{i_2} - v_{i'} - v_j \in G_2 - - v_i \in G_1$~~
~~$\Rightarrow$ atleast 2~~ $2 *$ no. of crossovers.
if Γ starts from G_1, one edge from $G_1 \to G_2 \Rightarrow$ edge G_2 & G_1 as Γ is closed
$\Rightarrow |E(\Gamma \cap S)|$ are always even

★ Fundamental cut set.

sp. tree T in connected graph G. Take a branch 'b' of T.
$S = \{b\} \to$ cut set of T.

★ Cut set containing exactly one branch of a tree T (and other chords) is fundamental cut set of G wrt T.

Th w.r.t a spanning tree T, a chord c_i that determines a fundamental circuit T' occurs in every fundamental cutset associated with branches in T' & no other.

Proof Given a spanning tree T in a connected graph G. Let c_i be the chord w.r.t T.
Let the fundamental circuit T' made by c_i be
$T' = \{b_1, b_2 \ldots b_k, c_i\}$ $b_i \in T \;\forall i = 1:k$
Every branch of T has a fundamental cut set associated

Let S_i be the fundamental cut set containing b_i
$S_i = \{b_i, c_1 c_2 \ldots c_p\}$

Using theorem: Any cut set has even no. of edges common with a circuit in the given graph.
$\Rightarrow |T' \cap S_i| = \text{even}$
$b_i \in T' \cap S_i$
$\Rightarrow c_i \in T' \cap S_i$ as S_i contains only one branch, and T' contains only chord c_i and $|T' \cap S_i| = \text{even}$

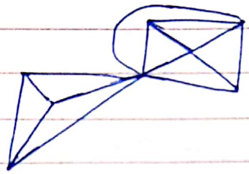
Considering S' {cut set not corresponding to T}
$\Rightarrow b_i \notin S' \;\forall i$
$\Rightarrow$ if $c_i \in S'$ $|T' \cap S'| = 1 \rightarrow$ contradiction
$\Rightarrow c_i \notin S'$
Hence Proved

Edge Connectivity
Given a graph G, each cut set of G consists of a certain no. of edges.
→ The no of edges in the smallest cut set is Edge Connectivity
→ Edge connectivity = min {cardinality (cut set)}.

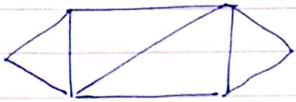
Vertex Connectivity (Connectivity)

The minimum no. of vertices whose removal from graph G, disconnects it.

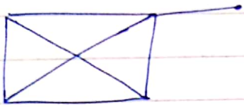
eg



(vertex connectivity) = 1
edge connectivity = 3



vertex = 2.
edge = 2.



vertex > 1
edge = 1

Separable Graph

A graph G is called separable if its vertex connectivity = 1 and the vertex removed to disconnect the graph is called the cut vertex. or Articulation Point

→ Connectivity (vertex) = 1
cut vertex → the 1 vertex to be removed

K connected graph

Graph with (vertex) connectivity = K

Max Flow Problem

It involves finding a feasible flow through a single source, single sink flow network that is maximum.

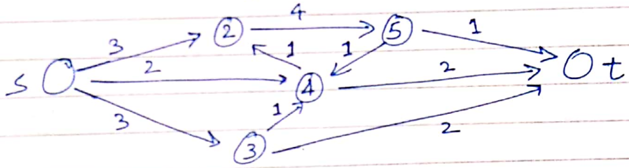
Def^n s: source   t: sink (destination)

An st-flow is an assignment of values to the edges in the network

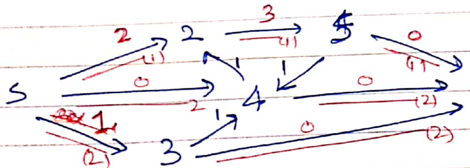
(i) capacity constraint
  0 ≤ edge's flow ≤ edge's capacity

(ii) equilibrium constraint (local)
  inflow = out flow at every vertex. {excluding s & t}

Ford Fulkerson Algorithm



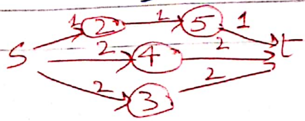
(i) Find an st path
(ii) determine the capacity of the path, Δ.
(iii) send Δ units of flow on the path
update the capacities of edges in original network.
Repeat above steps until no st path exists



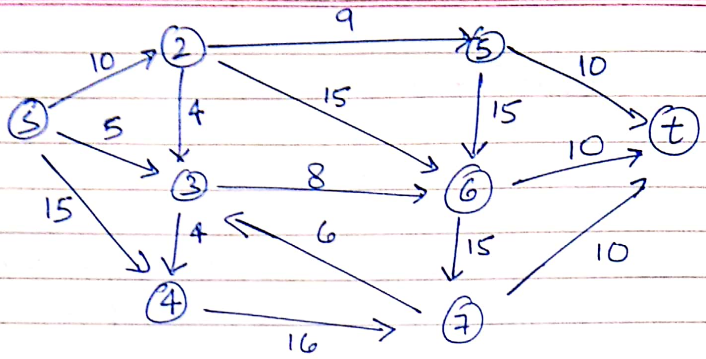
draw new graph at each step

$s \to 2 \to 5 \to t$ (1)
$s \to 4 \to t$ (2)
$s \to 3 \to t$ (2)

max flow $= 1+2+2 = 5$

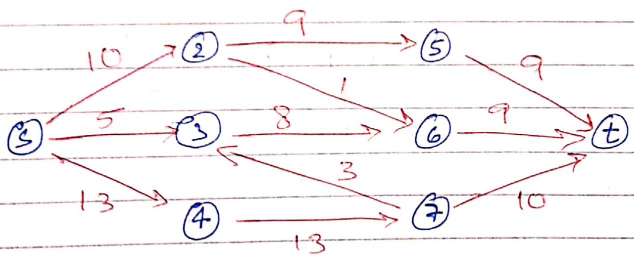


(Final flow diagram)
(use flow NOT capacity)



s − 2 − 5 − t (9)
s − 4 − 7 − t (10)
s − 3 − 6 − t (5)
s − 2 − ~~3~~ − 6 − t (1)
s − 4 − 7 − 3 − 6 − t (3)

Max flow = 28



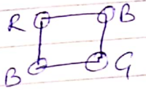
Coloring (or Vertex Coloring) (proper vertex coloring)

Let G be a graph. A proper vertex coloring of G is a labeling of the vertex set

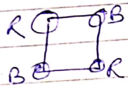
$$f : V(G) \to \underbrace{\{1, 2, \dots k\}}_{\text{labels}}$$

where the labels are called COLORS & no two adjacent vertices get the same color.

eg



3 colors



2 colors

applications → scheduling exams/classes simultaneously
determining class / division of vertices into groups.

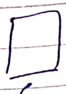
K-coloring

A k-coloring of a graph G is a coloring of G using K colors.
Such a graph is called k-colorable

Chromatic Number

Defined as the smallest K such that the graph G is K colorable, for a graph G.
Denoted by $\chi(G)$.

eg



⇒ 2 colorable, 3 colorable, 4 colorable
$\chi(G) = 2$

* $\chi(\text{Path}) = 2$
* $\chi(\text{cycle}) = \begin{Bmatrix} \chi(C_{2n}) = 2 \\ \chi(C_{2n+1}) = 3 \end{Bmatrix}$
* $\chi(K_n) = n$

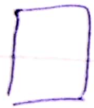
$\chi(G) = 1 \Rightarrow |E(G)| = 0$

...ing of G using

Such a graph is called k-colorable

Chromatic Number

Defined as the smallest k such that the graph is k colorable, for a graph G.
Denoted by $\chi(G)$.

e.g.  $\Rightarrow$ 2 colorable, 3 colorable, 4 colorable

G

$\chi(G) = 2$

* $\chi(\text{Path}) = 2$
* $\chi(\text{cycle}) = \begin{cases} \chi(C_{2n}) = 2 \\ \chi(C_{2n+1}) = 3 \end{cases}$
* $\chi(K_n) = n$
* $\chi(G) \leq |V(G)|$
* $H \subseteq G \Rightarrow \chi(H) \leq \chi(G)$

Subgraph

$\chi(G) = 1 \Rightarrow |E(G)| = 0$

Chromatic Partitioning

Let $G(V,E)$ is a graph. A k-coloring of G partitions V into k subsets → one for each color.

$V \to V_1, V_2 \ldots V_k$ where V_i is independent set $\forall i$.

independent set ⇒ No two vertices in the set are adjacent.

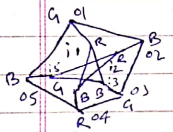
$V = V_1 \cup V_2 \ldots \cup V_k$

$V_i \cap V_j = \phi \quad i \neq j$.

eg G: Petersen Graph

$\chi(G) = 3$

⇒ 3 partitions



$V = \{i1, i2, o4\} \cup \{i3, i4, o2, o5\} \cup \{o1, o3, i5\}$

2-chromatic Graph $\{$all $G : \chi(G) = 2\}$

k-colorable ⇒ $k-1$ colorable (may)

k-chromatic ⇒ NOT $k-1$ colorable

Th Every tree with at least 2 vertices is 2-chromatic.

Let T be a tree with $|V(T)| \geq 2$.

Let 'v' be any arbitrary vertex in T. We call 'v' 'root'.

color v using (color 1).

$X = \{u \in V(T) \mid dist(root, u) = odd\}$ $Y = V(T) - X - \{root\}$

we color X using (color 2) and Y using (color 1).

Since there is a unique path b/w every pair of vertices, No ~~to~~ vertex gets two colors & the coloring is proper. i.e. No adjacent vertices get the same color.

∵ T is arbitrary, Hence every tree is 2-chromatic

Th If $\chi(G)=2 \iff G$ is bipartite.
(G is non-trivial)

Let G is bipartite. By defn?
$V(G) = V_1 \oplus V_2$
s.t $e \in E(G) \Rightarrow e(x,y) : x \in V_1, y \in V_2$
we color $v \in V_1$ using (color 1)
and $v \in V_2$ using (color 2)
$\Rightarrow G$ is 2-chromatic or $\chi(G) = 2$.

$\chi(G) = 2 \Rightarrow G$ is bipartite.
contraposition
If G is not bipartite $\Rightarrow \chi(G) > 2$.

A given graph G is bipartite $\iff G$ has no odd cycles.

G is not bipartite $\Rightarrow G$ has at least one odd cycle
consider H (the odd cycle) $\subseteq G$.
$\chi(G) \geq \chi(H) = 3$
$\chi(G) \geq 3$
$\chi(G) > 2$
$\Rightarrow \chi(G) = 2 \Rightarrow G$ is bipartite.

04/11/2020

Matching

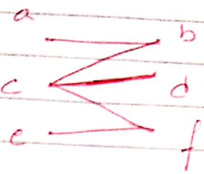
A subset of edges of a graph where no two edges are adjacent.

Maximal matching → No more edges can be added

Maximum matching → max number of edges.

Date : 04/11/20
Page No. 38

eg



matching {a—b} matching not maximal

{a—b, c—d, e—f} {a—b, c, d, e—f}

both are maximal matching

⇒ NOT unique.

{a—b, c—d, e—f} Maximum matching ⇒ one of maximal matching set.

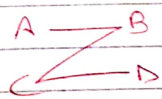
★ Maximum matching ⊆ maximal matching.

★ Matching number → no. of edges in ~~largest Maximal~~ maximum matching

eg {a—b, c—d, e—f} matching no = 3.

★ Perfect Matching – All the vertices are matched.

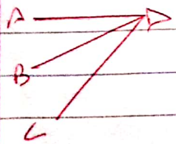
eg



{A—B, C—D} {A, B, C—B, D}

maximal

{A—B, C—D} → maximum, perfect.



⇒ NO perfect matching possible

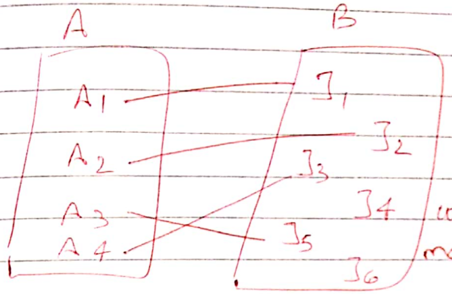
Complete Matching (for bipartite graph)

$f : A \to B$

Domain, codomain

All the vertices in the domain are matched.

eg



$V = A \oplus B$

complete matching

In a bipartite graph, one subset is domain, the other codomain. All the vertices in domain need to be matched [Resembles a function].

Number of complete matchings in $K_{n,n}$.

$n!$

~~ways to connect~~

$V(K_{n,n}) = \{a_1 \ldots a_n\} \oplus \{b_1 \ldots b_n\}$

ways to connect $a_1 = n$

$a_2 = n-1$

$a_n = 1$

$\Rightarrow$ total ways $= n \times (n-1) \ldots \times 1 = n!$

Number of Perfect matching $C_{2n} = 2$

$C_{2n} \to$ cycle of even length

Number of Perfect matchings in K_{2n}

$(2n-1)(2n-3)(2n-5) \cdots 1$

$= \dfrac{2n!}{2^n(n!)}$

Only one (trivial) matching in Null graph

$E = \phi$

$\Rightarrow M \subseteq E = \phi.$

No non-trivial matching as No edges altogether.