Cut-sets and cut-vertices

Cut-set

- In a connected graph G, a cut-set (cocycle) is a set of edges whose removal from G leaves G disconnected.

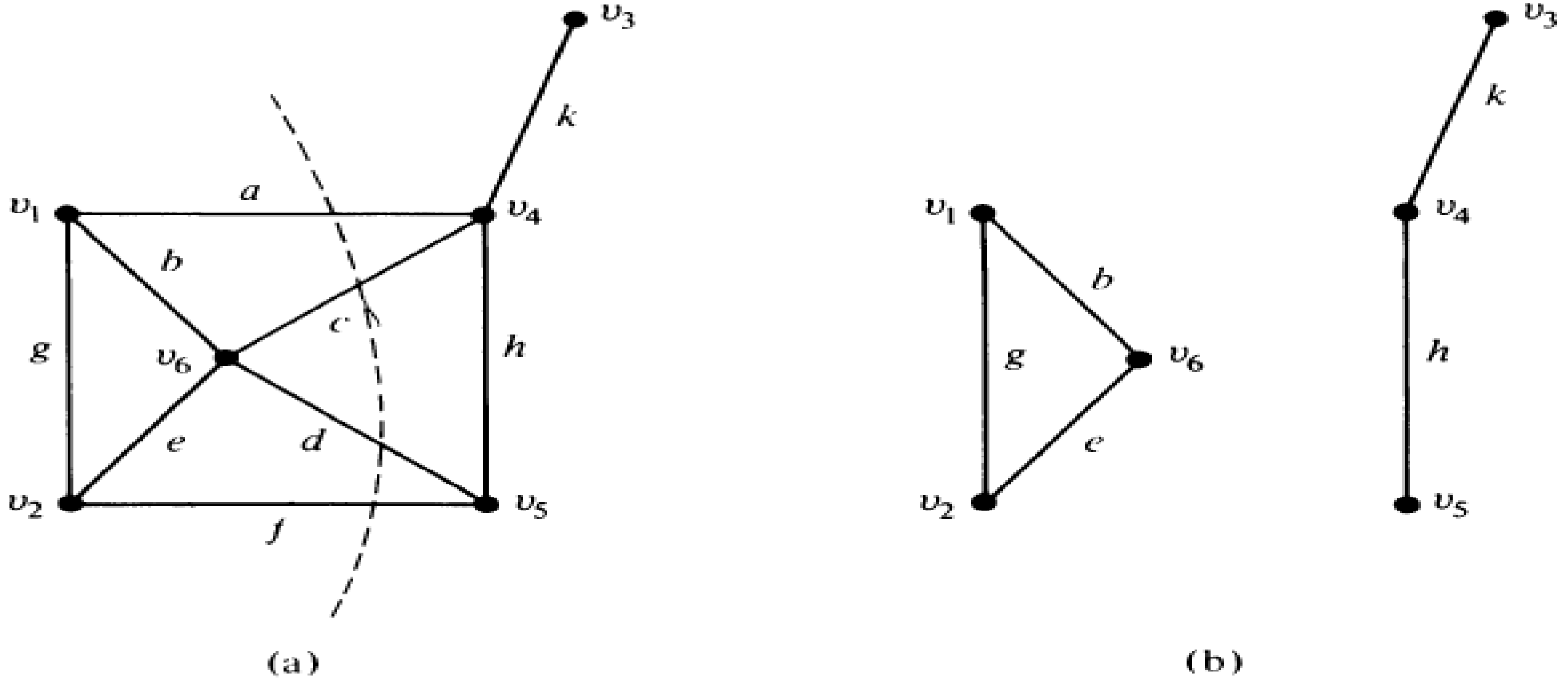


Fig. 4-1 Removal of a cut-set $\{a, c, d, f\}$ from a graph "cuts" it into two.

Cut-sets are {a,c,d,f}, {a,b,g}, {a,b,e,f} and {d,h,f}. What about edge {k} ? {a,c,h,d} is not a cut-set because one of its proper subset {a,c,h} is a cut-set.

- To emphasize the fact that no proper subset of a cut-set can be a cut-set, the cut-set is referred as a *minimal cut-set*, a *proper cut-set* or a *simple cut-set*.
- The cut-set always cuts the graph into two. It is defined as a minimal set of edges in a connected graph whose removal reduces the rank of the graph by one.
- Rank of the graph in Fig. 4.1(b) is ___4___ which is less than that of graph in 4.1 (a) ___5___ .

connected

for a graph G, which contains n vertices, the rank of G is

rank = n-1

n vertices → 2 subsets

V_1 = 1 vertex , V_2 = n−1 vertices

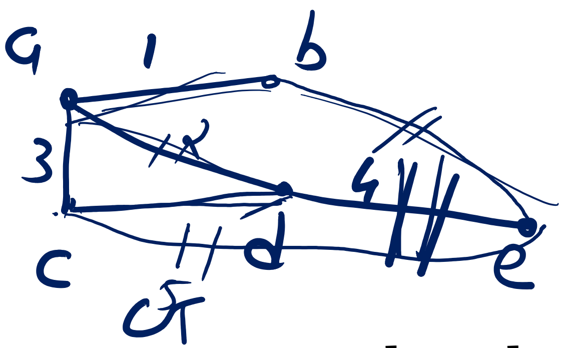
- Another way of looking at a cut-set is : if we partition all the vertices of a connected graph G into two mutually exclusive subsets, a cut-set is a minimal number of edges whose removal from G destroys all paths between these two sets of vertices.
- True or False: every edge of a tree is a cut-set. True

  branch
- Applications: communication and transportation networks, which help to identify the bottleneck of the network.

Properties of a cut-set

- Consider a spanning tree T in a connected graph G and an arbitrary cut-set S in G.
- Is it possible for S not to have any edge in common with T?
- No, otherwise removal of the cut-set S from G would not disconnect the graph.
- **Theorem 4.1**: Every cut-set in a connected graph G must contain at least one branch of every spanning tree G.

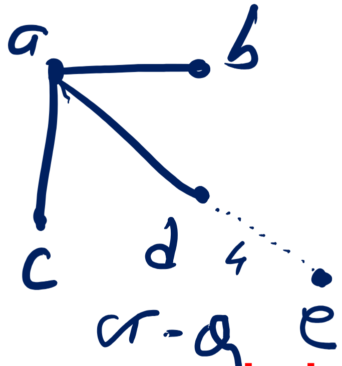
of



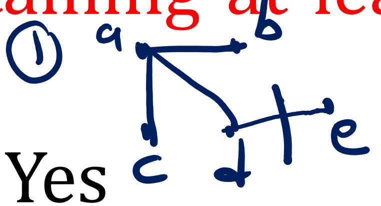
$Q = \{e\}$ 4

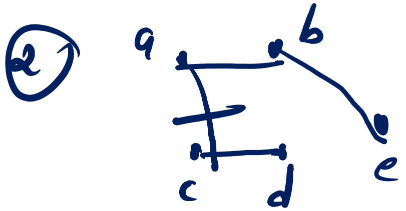
$V_1 = \{4\}$

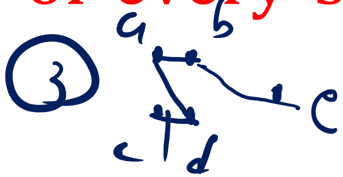
$\underline{G - Q}_{V_2} = \{1, 2, 3\}$



- Will the converse also be true? That is, will any minimal set of edges containing at least one branch of every spanning tree be a cut-set?







$Q: \{4, 3, 5\}$

- Answer is: Yes
- Prove. graph G, Q is a minimal set which contains atleast one branch of every spanning trees of G.

graph G-Q is a subgraph, which is disconnected.

Does G-Q has any spanning tree? No

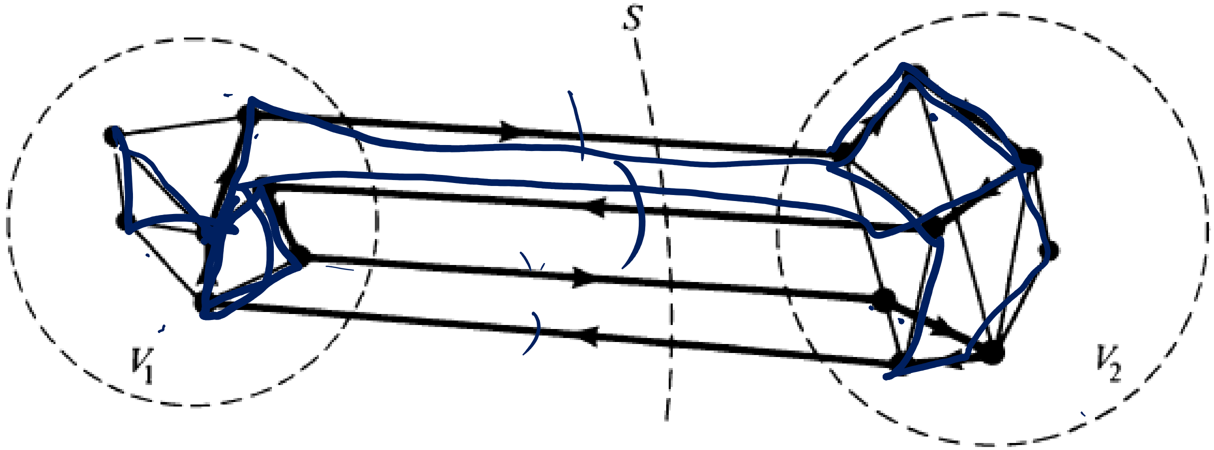
If any edge from Q is returned to G,

$\underline{G - Q + e}$ graph ⇒ connected

- **Theorem 4.2:** In a connected graph G, any minimal set of edges containing at least one branch of every spanning tree of G is a cut-set.
- **Theorem 4.3:** Every circuit has an even number of edges in common with any cut-set.

Find out any circuit Γ ∈ all vertices of (V_1 or V_2)

$N(S \cap \Gamma) = 0$



Circuit Γ shown in heavy lines, and is traversed along the direction of the arrows

Fig. 4-2 Circuit and a cut-set in G.

All cut-sets in a graph

- Cut-sets are used to identify weak spots in a communication net.
- This requires to list all cut-sets of the corresponding graph, and find which ones have the smallest number of edges.
- There are large number of cut-sets in a small graph, like 4.1, which requires a systematic method of generating all relevant cut-sets.

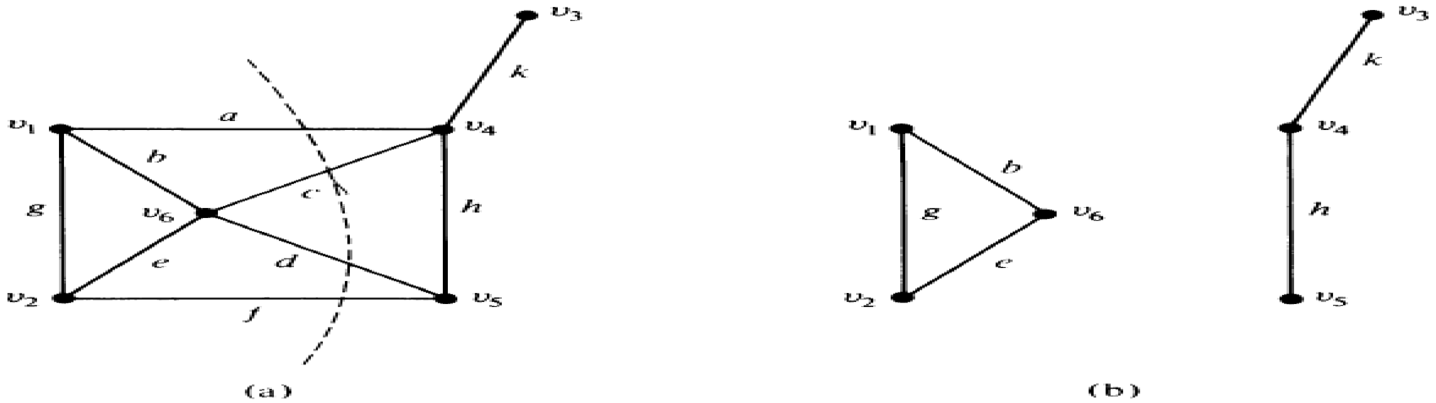


Fig. 4-1 Removal of a cut-set $\{a, c, d, f\}$ from a graph "cuts" it into two.

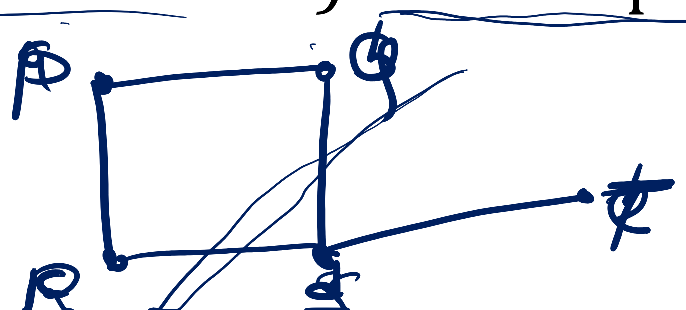
- How to find the circuits in a graph?

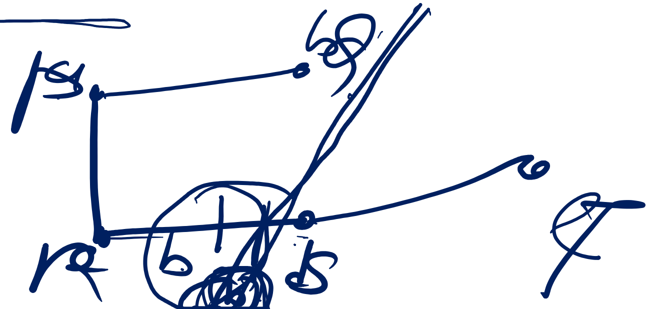
  simply find fundamental circuits & other circuits are found out using the combination of fundamental circuits.

- The strategy to find cut-sets is same.
- Just as a spanning tree is essential for defining a set of fundamental circuits, it is also essential for a set of *fundamental cut-sets*.

Fundamental cut-sets

- Consider a spanning tree T of a connected graph G. Take any branch *b* in T. Since {*b*} is a cut-set in T, {*b*} partitions all vertices of T into two disjoint sets – one at each end of *b*. Consider the same partition of vertices in G, and the cut-set S in G that corresponds to this partition.
- Cut-set S will contain only one branch b of T, and the rest (if any) of the edges in S are chords with respect to T. Such a cut-set S containing exactly one branch of a tree T is called a *fundamental cut-set (basic cut-set)* with respect to T.





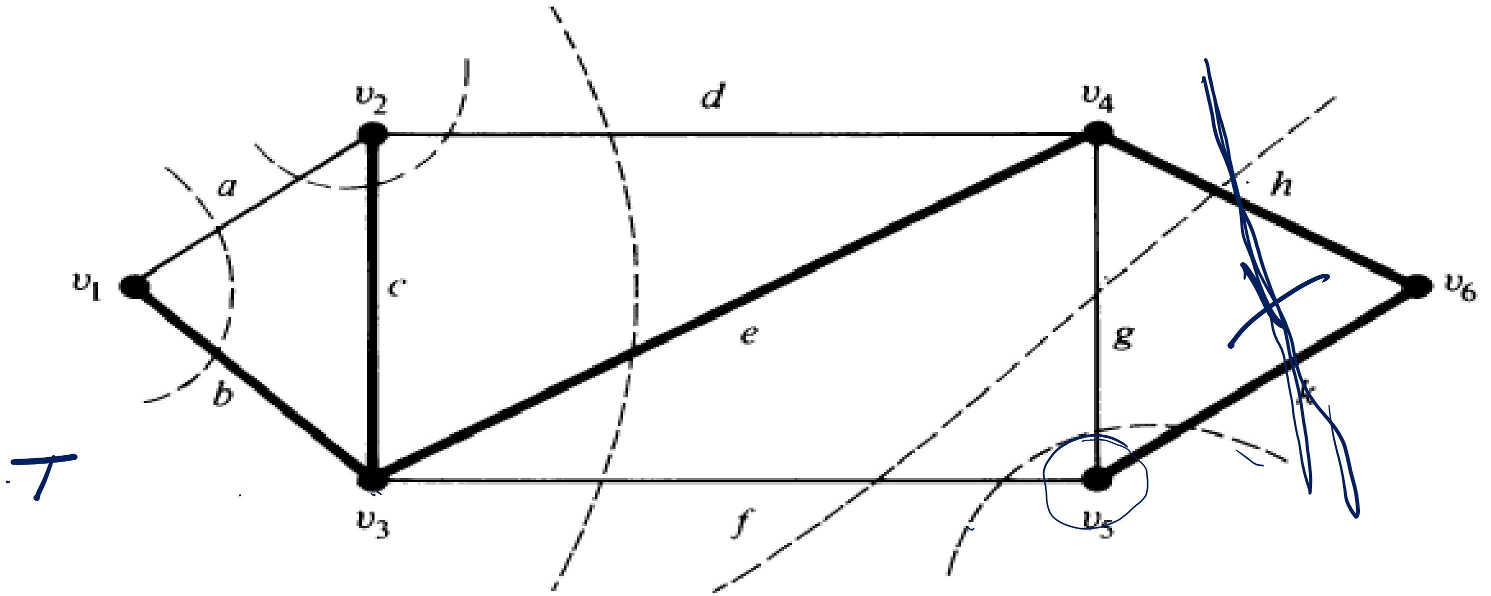
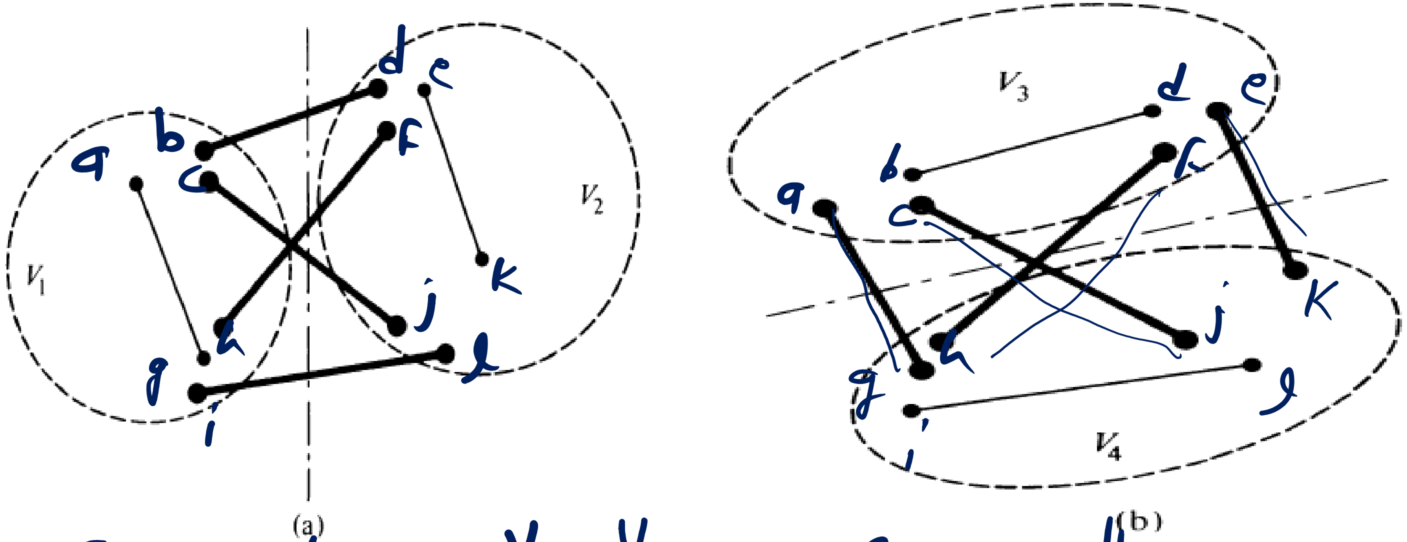


Fig. 4-3 Fundamental cut-sets of a graph.

- Just as every chord of a spanning tree defines a *unique* fundamental circuit, every branch of a spanning tree defines a *unique* fundamental cut-set. With respect to that spanning tree
- The fundamental cut-set has meaning only with respect to a given spanning tree.

How to obtain other cut-sets from a given set of cut-sets.

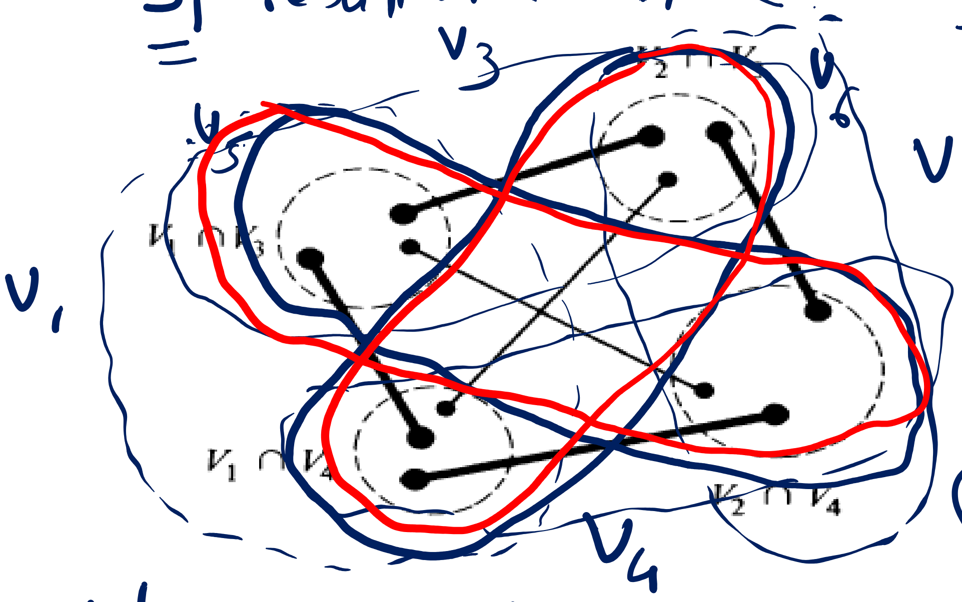
- **Theorem 4.4**: The ring sum of any two cut-sets in a graph is either a third cut-set or an edge disjoint union of cut-sets.



$S_1 : V_1 \cap V_2 = \phi$ & $V_1 \cup V_2 = V$

$S_2 : V_3 \cap V_4 = \phi$ & $V_3 \cup V_4 = V$

S_1 results in V_1, V_2 $\quad$ S_2 results in V_3 & V_4



$$\underbrace{(V_1 \cap V_4) \cup (V_2 \cap V_3)}^{V_1 \oplus V_3} \Rightarrow V_5$$

$$\underbrace{(V_1 \cap V_3) \cup (V_2 \cap V_4)}_{V_2 \oplus V_3} \Rightarrow V_6$$

No other edge joins vertices of V_5 & V_6

1) $S_1 \oplus S_2$ results into the set of edges which joins vertices of V_5 & V_6.

So, the new cut set resulted from $S_1 \oplus S_2 = S'$.
gives two subsets V_5 & V_6, where

$$V_5 \cap V_6 = \phi \quad \& \quad V_5 \cup V_6 = V$$

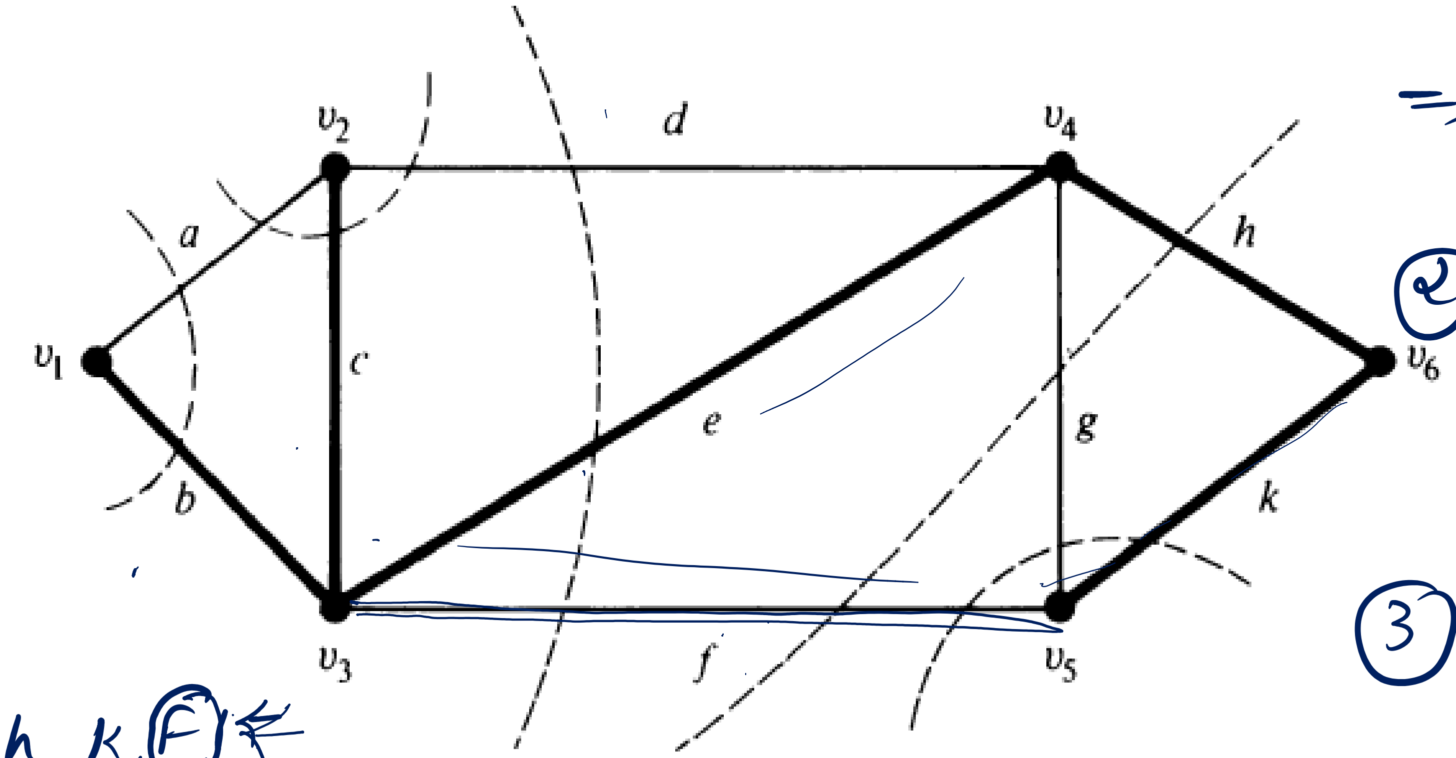
S' is a cutset if the subgraphs containing V_5 & V_6, remains connected if $\forall$ after removal of edges in s'.
otherwise s' is an edge-disjoint union of cut sets.

Example $T = \{b, c, e, h, k\}$

① $\{d, e, f\} \oplus \{f, g, h\}$
$= \{d, e, g, h\}$
$\Rightarrow$ is this the fundamental cut-set?

② $\{a, b\} \oplus \{b, c, e, f\}$
$= \{a, c, e, f\}$

③ $\{d, e, g, h\} \oplus \{f, g, k\}$
$= \{d, e, f, h, k\}$
$= \{d, e, f\} \cup \{h, k\}$



$\{e, h, k, f\}$

Fig. 4-3 Fundamental cut-sets of a graph.

- We cannot start with any two cut-sets in a given graph and obtain all its cut-sets by this method.
- What is a minimal set of cut-sets from which we can obtain every cut-set of G by taking ring sum? – Answer: set of all fundamental cut-sets with respect to a given spanning tree.

Fundamental Circuits and Cut-sets

- Consider a spanning tree T in a given connected graph G. Let c_i be a chord with respect to T, and let the fundamental circuit made by c_i be called Γ, consisting of k branches $b_1, b_2, \ldots, b_k$ in addition to the chord c_i, that is $\{b_1, b_2 \ldots b_k\}$ w.r.t. c_i is this set contain all the branches of T & No

c_i, $\{b_{k+1}, \ldots b_m\}$

$\Gamma = \{c_i, b_1, b_2, \ldots, b_k\}$ is a fundamental circuit with respect to T.

- Every branch of any spanning tree has a fundamental cut-set associated with it. Let S_1, be the fundamental cut-set associated with b_1, consisting of q chords in addition to the branch b_1, that is

$S_1 = \{b_1, c_1, c_2, \ldots, C_q\}$ is a fundamental cut-set with respect to T.

w.r.t. b_1 c_i is in c_1, c_2, c_i, c_q

- According to theorem 4.3, there must be an even number of edges common to Γ and S_1, and there is only one other edge in Γ (which is c_i) that can possibly also be in S_1.
- Therefore, there must be two edges b_1 and c_i common to Γ and S_1. Thus, the chord c_i is one of the chords $c_1, c_2, \ldots, c_q$.
- The same argument holds for fundamental cut-sets associated with $b_2, b_3, \ldots, b_k$. Therefore, the chord c_i is contained in every fundamental cut-set associated with branches of Γ.

Spanning tree T contains m branches $b_1, b_2, \_\_\_ b_m$

fundamental circuit T $(b_1, b_2 \_\_\_ b_k)$ $b_k, b_{k+1} \_\_\_ b_m$

- Is it possible for the chord c_i to be in any other fundamental cut-set S' (with respect to T) besides those associated with b_1, b_2, ..., b_k?
- Answer: No
- Otherwise (since none of the branches in Γ are in S'), there would be only one edge c_i common in S1 and Γ, a contradiction to Theorem 4.3.
- **Theorem 4.5**: With respect to a given spanning tree T, a chord c_i, that determines a fundamental circuit Γ occurs in every fundamental cut-set associated with the branches in Γ and in no other.

Example

circuit = {c, d, e}
cut set w.r.t c = {a, c, d}
w.r.t. e = {d, e, f}

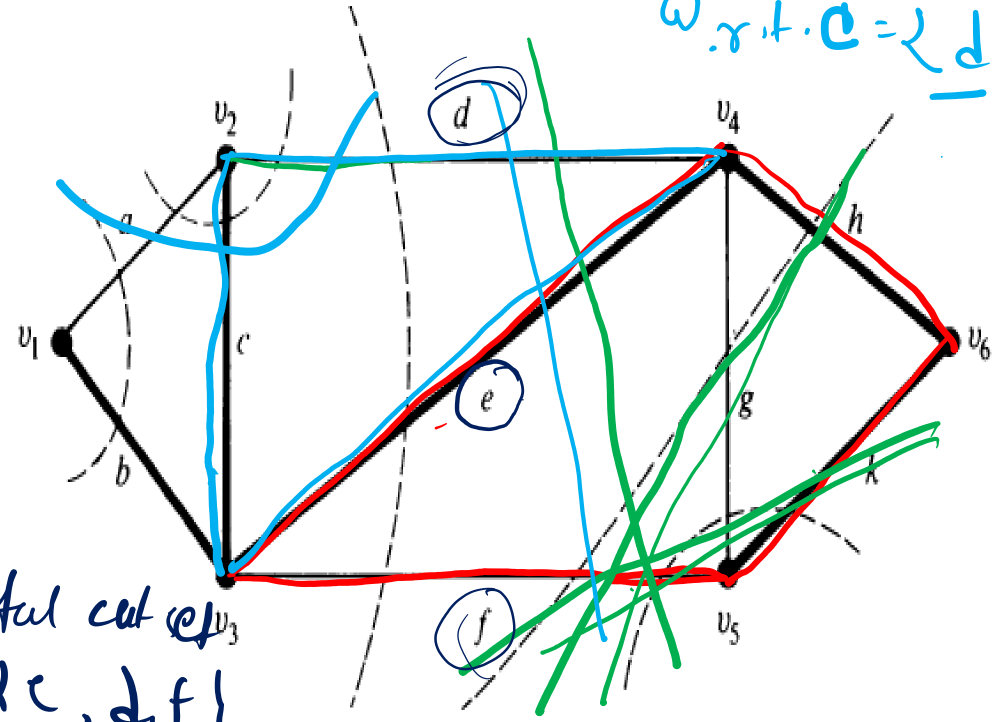


Fig. 4-3 Fundamental cut-sets of a graph.

fundamental cut set {c, d, f}

circuit w.r.t. d = {d, c, e}
f = {e, h, k, f}

- Consider the spanning tree {*b*,*c*,*e*,*h*,*k*} and the fundamental circuit made by chord f is {*f*,*e*,*h*,*k*}.
- The three fundamental cut-sets determined by the three branches *e*, *h* and *k* are:

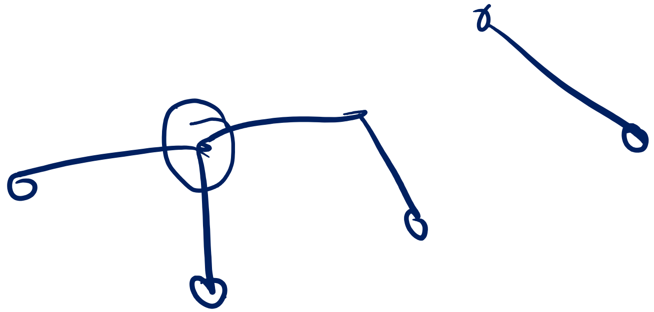
w.r.t. e = {d, e, f}
w.r.t h = {f, g, h}
w.r.t. k = {f, k}

Chord *f* occurs in each of these three fundamental cut-sets, and there is no other fundamental cut-set that contains *f*.

- The converse of Theorem 4.5 is also true.
- **Theorem 4.6**: With respect to a given spanning tree T, a branch b_i that determines a fundamental cut-set S is contained in every fundamental circuits associated with the chords in S, and in no other.
- Self study: Proof of Theorem 4.6.

Connectivity and Separability

- Edge Connectivity: Each cut-set of a connected graph G consists of a certain number of edges. The number of edges in the smallest cut-set (cut-set with fewest number of edges) is defined as the edge connectivity of G. In other words, the edge connectivity of a connected graph can be defined as the minimum number of edges whose removal reduces the rank of the graph by one.
- The edge connectivity of a tree is one.

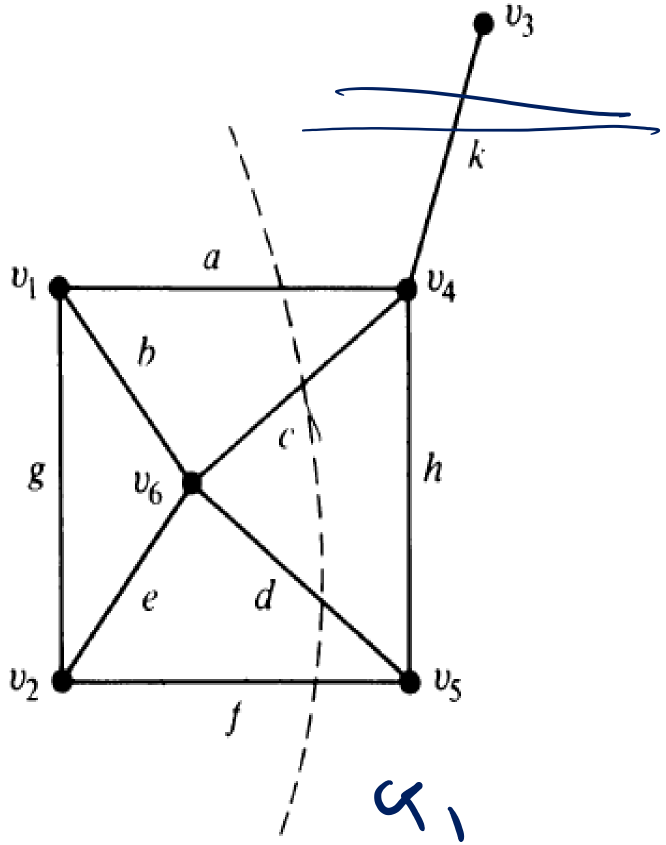


- Vertex connectivity: Vertex connectivity of a connected graph G is defined as the minimum number of vertices whose removal from G leaves the remaining graph disconnected.
- The vertex connectivity of a tree is one.
- The vertex connectivity is meaningful only for graphs that have three or more vertices and are not complete.
- Note: The edge and vertex connectivity of a disconnected graph is zero.
- What is the vertex connectivity of a complete graph with 5 vertices?

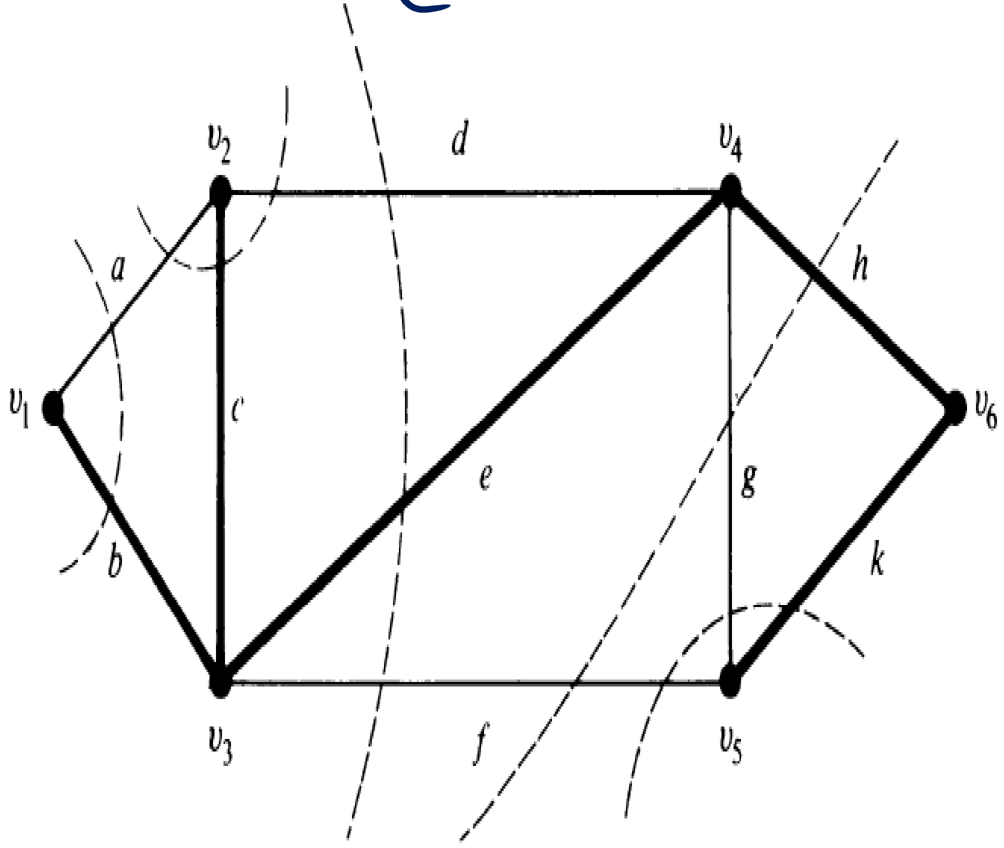
Edge and vertex connectivity Examples

edge connectivity = 1   2

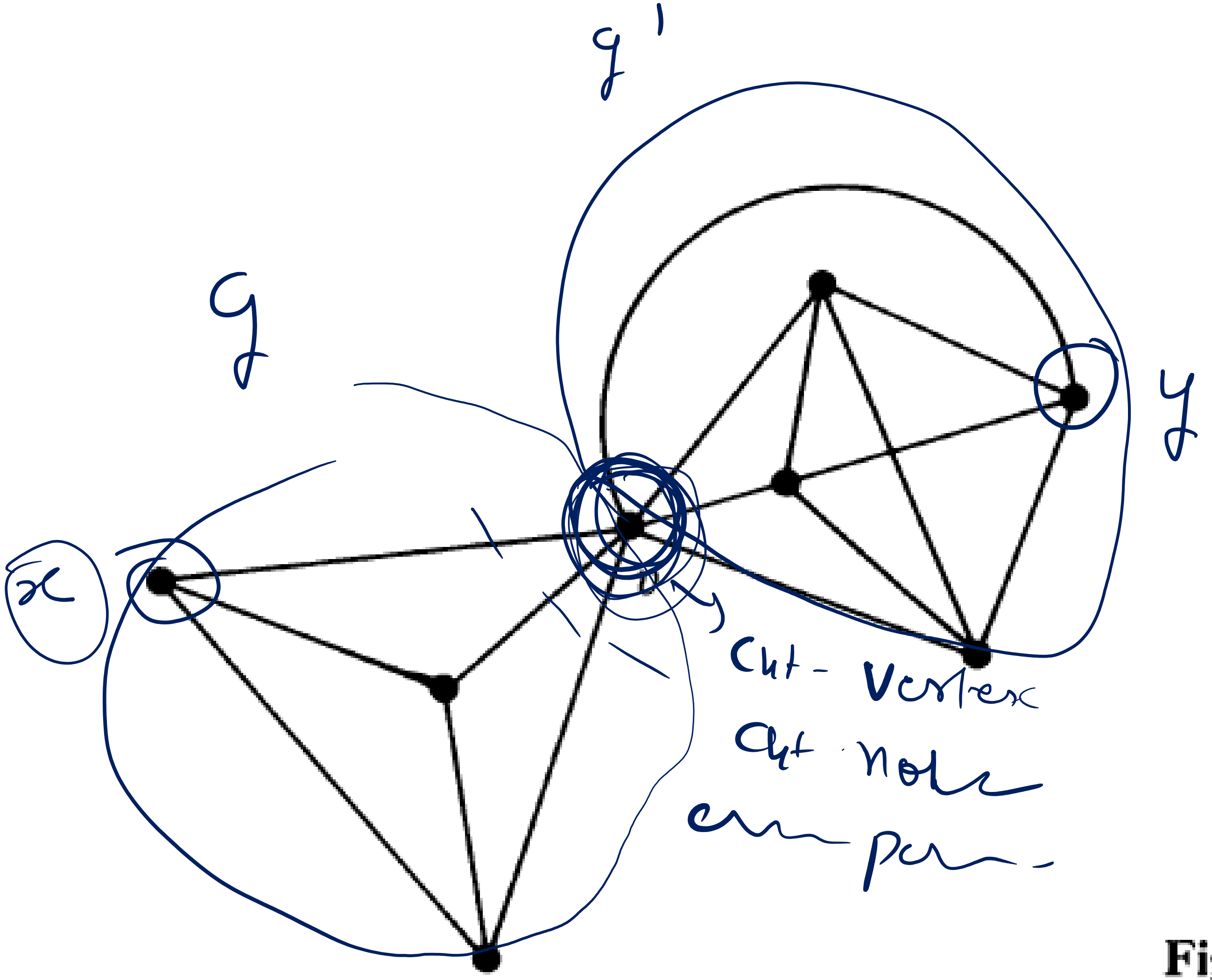
vertex connectivity = 1   2



4.1 (a) G_1



4.3 G_2.



edge connectivity = 3

vertex connectivity = 1

Fig. 4-5 Separable graph.

Separable Graph:

Karger's algorithm
min cut edge con

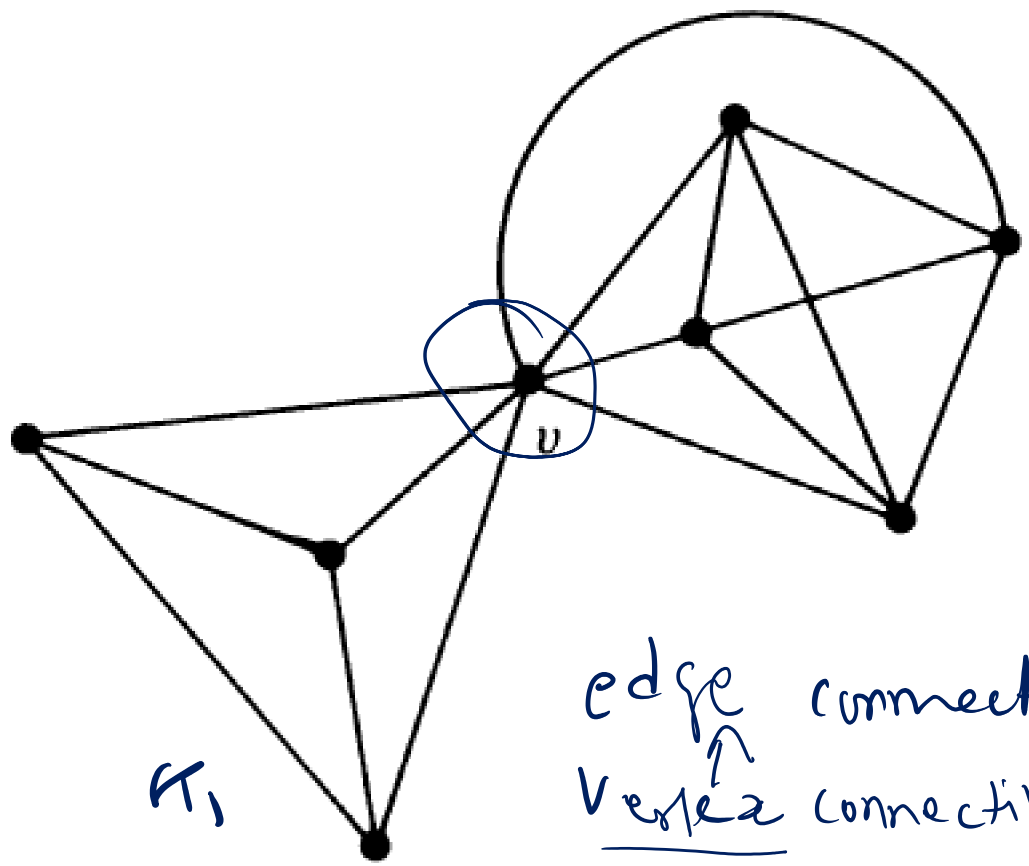
Separable Graph: A connected graph is said to be *separable* if its vertex connectivity is one. All other connected graphs are called *nonseparable*. An equivalent definition is that a connected graph G is said to be separable if there exists a subgraph g in G such that $\bar{g}$ (the complement of g in G) and g have only one vertex in common. That these two definitions are equivalent can be easily seen (Problem 4-7). In a separable graph a vertex whose removal disconnects the graph is called a *cut-vertex*, a *cut-node*, or an *articulation point*. For example, in Fig. 4-5 the vertex v is a cut-vertex, and in Fig. 4-1(a) vertex v_4 is a cut-vertex. It can be shown (Problem 4-18) that in a tree every vertex with degree greater than one is a cut-vertex. Moreover:

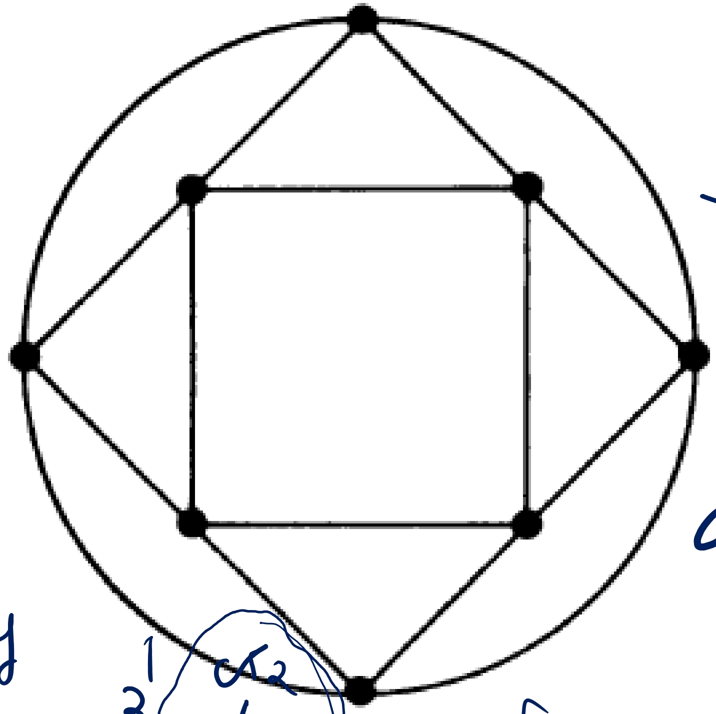
THEOREM 4-7

A vertex v in a connected graph G is a cut-vertex if and only if there exist two vertices x and y in G such that every path between x and y passes through v.

An Application: Suppose we are given n stations that are to be connected by means of e lines (telephone lines, bridges, railroads, tunnels, or highways) where $e \geq n - 1$. What is the best way of connecting? By "best" we mean that the network should be as invulnerable to destruction of individual stations and individual lines as possible. In other words, construct a graph with n vertices and e edges that has the maximum possible edge connectivity and vertex connectivity.

Two graphs with 8 vertices and 16 edges





		G_2
edge connectivity	3	4
Vertex connectivity	1	4

- What is the highest vertex and edge connectivity we can achieve for a graph with n vertices and e edges?
- **Theorem 4.8**: The edge connectivity of a graph G cannot exceed the degree of the vertex with the smallest degree in G.

Proof: Let vertex v_i be the vertex with the smallest degree in G. Let $d(v_i)$ be the degree of v_i. Vertex v_i can be separated from G by removing the $d(v_i)$ edges incident on vertex v_i. Hence the theorem. ■

- **Theorem 4.9**: The vertex connectivity of any graph G can never exceed the edge connectivity of G.

- COROLLARY: Every cut-set in a nonseparable graph with more than two vertices contains at least two edges.
- **Theorem 4.10**: The maximum vertex connectivity one can achieve with a graph G on n vertices and e edges (e≥ n-1) is the integral part of the number 2e/n ; $\left\lfloor \frac{2e}{n} \right\rfloor$.

- K-connected graph: A graph G is said to be k-connected if the vertex connectivity of G is k.
- Is a 1-connected graph is the same as a separable graph?
- **Theorem 4.11**: A graph G is k-connected if and only if every pair of vertices in G is joined by k or more paths that do not intersect and at least one pair of vertices is joined by exactly k nonintersecting paths.
- **Theorem 4.12**: The edge connectivity of a graph G is k if and only if every pair of vertices in G is joined by k or more edge-disjoint paths, and at least one pair of vertices is joined by exactly k edge disjoint paths.

Network Flow

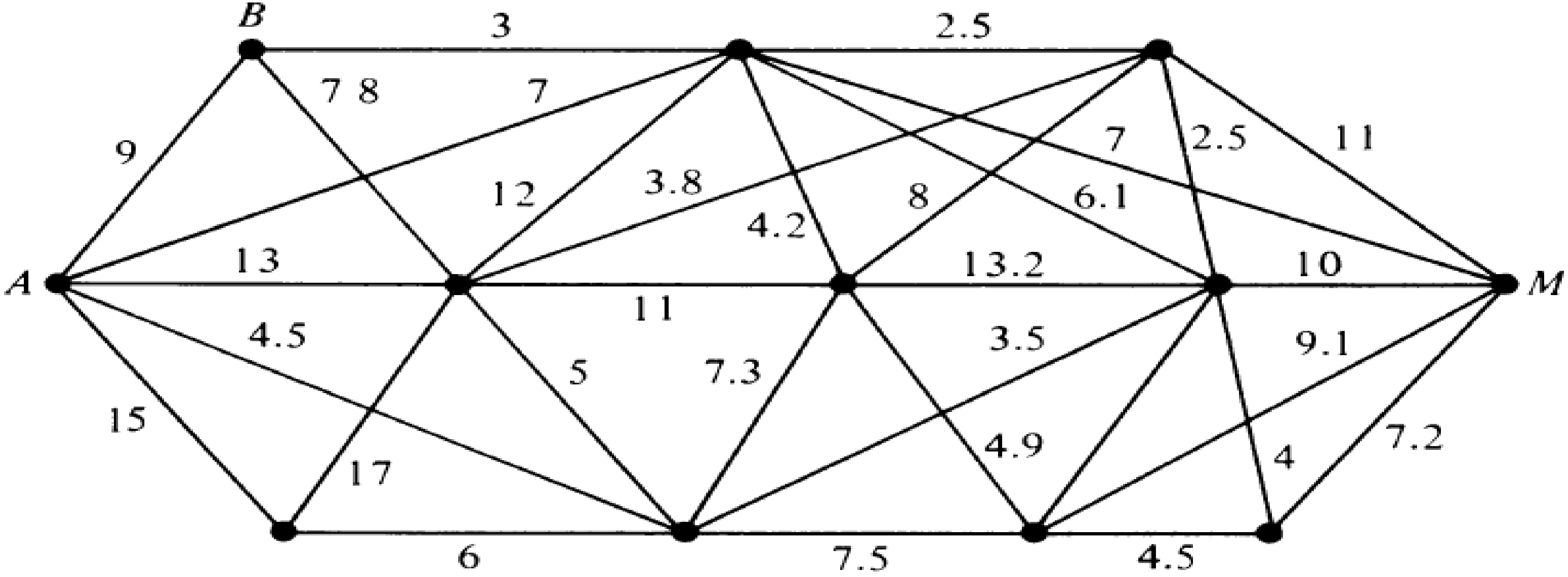


Fig. 4-7 Graph of a flow network.

- **Assumptions**
  - At each intermediate vertex the total rate of commodity entering is equal to the rate of leaving.
  - The flow through a vertex is limited only by the capacities of the edges incident on it.
  - Lines(edges) are lossless.
- **Questions**
  - What is the maximum flow possible through the network between a specified pair of vertices?
  - How do we achieve this flow?

- A *cut-set with respect to a pair of vertices a and b* in a connected graph G puts a and b into two different components.
- The capacity of a cut-set S in a weighted connected graph G (in which the weight of each edge represents its flow capacity) is defined as the sum of the weights of all the edges in S.
- **Theorem 4.13**: The maximum flow possible between two vertices a and b in a network is equal to the minimum of the capacities of all cut-sets with respect to a and b.

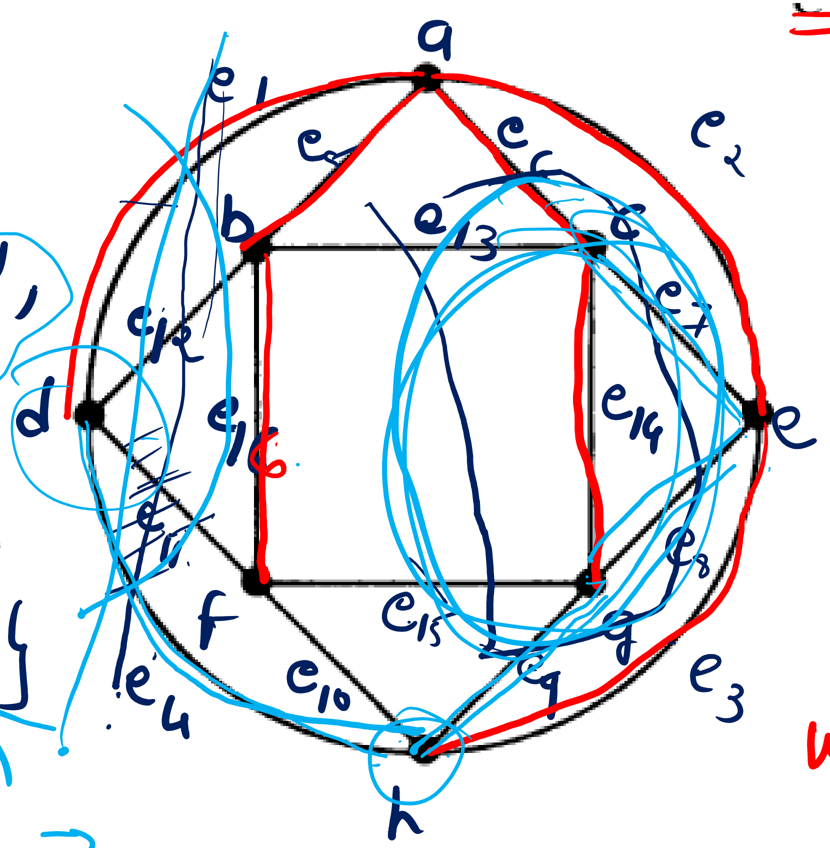
Pick out any spanning tree and list all seven fundamental cutsets w.r.t. the tree.

$T = \{e_1, e_2, e_3, e_5, e_6, e_{16}, e_{14}\}$

$e_1 \oplus e_6$

$= \{e_1, e_{12}, e_{11}, e_4, e_6, e_7, e_8, e_9, e_{15}, e_{13}\}$

$\Rightarrow$



3 connected com...

w.r.t. $e_1 = \{e_1, e_{12}, e_{11}, e_4\}$

w.r.t. $e_2 = \{e_2, e_7, e_8, e_9, e_{10}, e_4\}$

w.r.t. $e_3 = \{e\ , e_9, e_{10}, e_4\}$

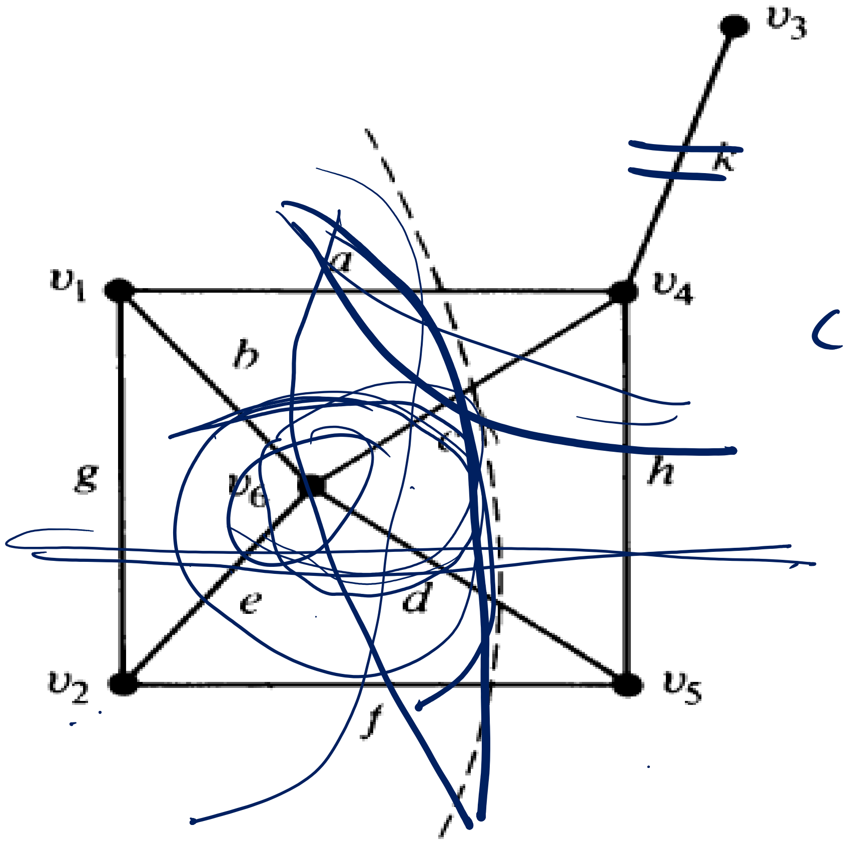
w.r.t. $e_5 = \{e_5, e_{13}, e_{15}, e_{10}, e_{11}, e_{12}\}$

w.r.t. $e_6 = \{e_6, e_7, e_8, e_9, e_{15}, e_{13}\}$

w.r.t. $e_{16} = \{e_{16}, e_{11}, e_{10}, e_{15}\}$

w.r.t. $e_{14} = \{e_{14}, e_8, e_9, e_{15}\}$

/i's list out all cut-sets w.r.t. the vertex pair V_2 & V_3.



Cutsets = $\{\underline{k}\}$ $V_a = \{v_1, v_2, \cancel{v_3}\ v_6, v_4, v_5\}$

$V_b = \{v_3\}$

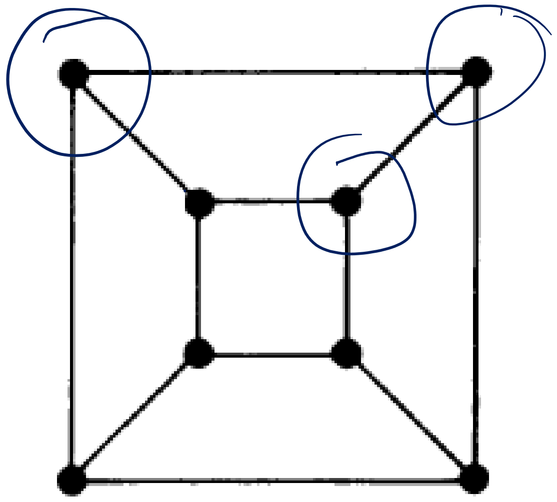
Cutsets = $\{\cancel{a, c,} d\}$, $\{a, c, h\}$,

$\{g, e, f\}$, $\{g, b, c, h\}$

$\{\cancel{b, e, d}\}$, $\{g, e, d, h\}$

$\{a, b, e, f\}$, $\{a, c, d, f\}$

What is the edge connectivity & Vertex connectivity ?



edge connu : 3

Vu.. = 3