

Part I

ME010101 Abstract
Algebra

Chapter 1

Module 1

Reading Material : [?, §11]

Definitions : $A \times B$ (§0.4), relation(§0.7), binary operation(§2.1), group(§4.1)

1.1 Direct Products and Finitely Generated Abelian Groups

Part II

ME010102 Linear Algebra

Chapter 2

Vector Spces

Chapter 3

Linear Transformations

Chapter 4

Determinants

Chapter 5

Elementary Canonical Forms

Part III

ME010103 Basic Topology

Chapter 6

Topological Spaces

6.1 Definition of a Topological Space

6.2 Examples of Topological Spaces

6.3 Bases and Sub-bases

6.4 Subspaces

Chapter 7

Basic Concepts

7.1 Closed Sets and Closure

7.2 Neighbourhoods, Interior and Accumulation Points

7.3 Continuity and Related Concepts

7.4 Making Functions Continuous, Quotient Spaces

Chapter 8

Spaces with Special Properties

8.1 Smallness Conditions on a Space

8.2 Connectedness

8.3 Local Connectedness and Paths

Chapter 9

Separation Axioms

9.1 Hierarchy of Separation Axioms

Part IV

ME010103 Real Analysis

Chapter 10

Functions of Bounded Variation & Rectifiable Curves

Chapter 11

The Riemann Stieltjes Integral

Chapter 12

Sequence & Series of Functions

Chapter 13

Weierstrass Approximation & Some Special Functions

Part V

Graph Theory

Chapter 14

Graphs

Chapter 15

Directed Graphs

Chapter 16

Connectivity

Chapter 17

Trees

Chapter 18

Eulerian & Hamiltonian Graphs

Chapter 19

Graph Colorings

Chapter 20

Planarity

Chapter 21

Spectrum of Graphs

Part VI

ME010201 Advanced Abstract Algebra

Part VII

**ME010202 Advanced
Topology**

Part VIII

ME010204 Complex Analysis

Part IX

ME010205 Measure & Integration

Part X

ME010301 Advanced Complex Analysis

Part XI

ME010302 Partial Differential Equations

Part XII

ME010303 Multivariate Calculus & Integral Transforms

Part XIII

**ME010304 Functional
Analysis**

Part XIV

ME010305 Optimization Technique

Part XV

ME010401 Spectral Theory

Part XVI

ME010402 Analytic Number Theory

Part XVII

**ME800401 Differential
Geometry**

Part XVIII

**ME800402 Algorithmic
Graph Theory**

Chapter 22

An Introduction to Graphs

- 22.1 What is a Graph ?
- 22.2 The Degree of a Vertex
- 22.3 Isomorphic Graphs
- 22.4 Subgraphs
- 22.5 Degree Sequences
- 22.6 Connected Graphs
- 22.7 Cut-Vertices and Bridges
- 22.8 Special Graphs
- 22.9 Digraphs

Chapter 23

An Introduction to Algorithms

23.1 Algorithmic Complexity

23.2 Search Algorithms

23.3 Sorting Algorithms

23.5 Greedy Algorithms

23.6 Representing Graphs in a Computer

Chapter 24

Trees

24.1 Properties of Trees

24.2 Rooted Trees

24.3 Depth-First Search

24.4 Depth-First Search : A Tool for Finding
Blocks

24.5 Breadth-First Search

24.6 The Minimum Spanning Tree Problem

Chapter 25

Paths and Distance in Graphs

25.1 Distance in Graphs

25.2 Distance in Weighted Graphs

25.3 The Center and Median of a Graph

25.4 Activity Digraphs and Critical Paths

Chapter 26

Networks

26.1 An Introduction to Networks

Definitions 26.1. A **network** N is a digraph D with two special vertices source s and sink t together with a capacity function $c : E(D) \rightarrow \mathbb{Z}$ such that for every arc $a = (u, v)$ of the digraph, $c(u, v)$ is non-negative.

Remark. *Mathematical Modeling using Network,*

1. There is no restriction on indegree/outdegree of source/sink vertices of the digraph D of a network N .
2. Applications of Network : Transportation problem.

$c(u, v)$ is the capacity of the arc (u, v) of D

$N^+(x) = \{y \in V(D) : (x, y) \in E(D)\}$ is the out-neighbourhood of x .

$N^-(x) = \{y \in V(D) : (y, x) \in E(D)\}$ is the in-neighbourhood of x .

Definitions 26.2. A **flow f in a network N** is function $f : E(D) \rightarrow \mathbb{Z}$ such that 1. each edge satisfies capacity constraint and 2. each vertex except source and sink satisfies conservation equation.

capacity constraint

$$0 \leq f(a) \leq c(a) \text{ for every arc } a \in V(D) \quad (26.1)$$

conservation equation

$$\sum_{y \in N^+(x)} f(x, y) = \sum_{y \in N^-(x)} f(y, x), \quad \forall \text{ vertex } x \in V(D) - \{s, t\} \quad (26.2)$$

net flow out of x

$$\sum_{y \in N^+(x)} f(x, y) - \sum_{y \in N^-(x)} f(y, x)$$

net flow into x

$$\sum_{y \in N^-(x)} f(y, x) - \sum_{y \in N^+(x)} f(x, y)$$

Definitions 26.3. The **flow f in a network N** is the net flow out of source s .

Remark. 1. net flow out of/into $x \in V(D) - \{s, t\}$ is zero.

2. Without loss of generality¹, underlying digraph is always assymetric.

$(X, Y) = \{(x, y) \in E(D) : x \in X, y \in Y\}$.

Let X, Y be non-empty subsets of $V(D)$ such that X, Y are disjoint. Then (X, Y) is the set of all arcs from X to Y .

flow from X to Y is the sum of flow on each arc in (X, Y)

$$f(X, Y) = \sum_{(x, y) \in (X, Y)} f(x, y) \quad (26.3)$$

capacity of the partition (X, Y) is the total capacity of arcs in (X, Y)

$$c(X, Y) = \sum_{(x, y) \in (X, Y)} c(x, y) \quad (26.4)$$

cut Let $P \subset V(D)$ such that $s \in P$ and $t \notin P$ and $\bar{P} = V(D) - P$, then (P, \bar{P}) is a cut.

flow from P to \bar{P} is the sum of flow on each arc in (P, \bar{P}) .

$$f(P, \bar{P}) = \sum_{(x, y) \in (P, \bar{P})} f(x, y) \quad (26.5)$$

flow from \bar{P} to P is the sum of flow on each arc in (\bar{P}, P)

$$f(\bar{P}, P) = \sum_{(x, y) \in (\bar{P}, P)} f(x, y) \quad (26.6)$$

capacity of the cut (P, \bar{P}) is the total capacity of the arcs in (P, \bar{P})

$$c(P, \bar{P}) = \sum_{(x, y) \in (P, \bar{P})} c(x, y) \quad (26.7)$$

Theorem 26.4. For any cut (P, \bar{P}) , the flow in N is $f(N) = f(P, \bar{P}) - f(\bar{P}, P)$.

Synopsis. The net flow out of source s is the flow $f(N)$ in the network N . Let (P, \bar{P}) be a cut of N , then $s \in P$ and $t \notin P$. Suppose $P = \{s\}$, then the theorem is true. Suppose P is not singleton, then for each vertex $x \in P$, $x \neq s$, the net flow out of x is zero by flow conservation equation. And flow between vertices in P cancels out each other. Thus adding net flow out of each vertex in P , will be same as the net flow out of source which is the flow in the network, $f(N)$.

¹If underlying digraph of a network is symmetric, then by replacing an arc (u, v) with a new vertex w and two arcs $(u, w), (w, v)$ gives an assymetric digraph.[?]pp.131

Proof.

$$\text{Flow, } f = \sum_{y \in N^+(s)} f(s, y) - \sum_{y \in N^-(s)} f(y, s) \quad (26.8)$$

By conservation equation, we have $\forall x \in P, x \neq s$,

$$\sum_{y \in N^+(x)} f(x, y) - \sum_{y \in N^-(x)} f(y, x) = 0 \quad (26.9)$$

By above equations,

$$\begin{aligned} \text{Flow, } f &= \sum_{x \in P} \sum_{y \in N^+(x)} f(x, y) - \sum_{x \in P} \sum_{y \in N^-(x)} f(y, x) \\ &= \sum_{(x, y) \in (P, \bar{P})} f(x, y) - \sum_{(y, x) \in (\bar{P}, P)} f(y, x) \end{aligned} \quad (26.10)$$

□

Corollary 26.4.1. *Flow cannot exceed the capacity of any cut (P, \bar{P}) . Further, $f(N) \leq \min c(P, \bar{P})$.*

Synopsis. *Let (P, \bar{P}) be a cut in network N , then by theorem the flow $f(N) = \text{flow from } P \text{ to } \bar{P} - \text{flow from } \bar{P} \text{ to } P$. Since the flow from \bar{P} to P is non-negative, $f(N) \leq \text{flow from } P \text{ to } \bar{P}$. Clearly, $f(x, y) \leq c(x, y)$ by the capacity constraint. Thus $f(N) \leq f(P, \bar{P}) \leq c(P, \bar{P}) \leq \min c(P, \bar{P})$.*

Proof.

$$\begin{aligned} f(N) &= \sum_{(x, y) \in (P, \bar{P})} f(x, y) - \sum_{(y, x) \in (\bar{P}, P)} f(y, x) \\ &\leq \sum_{(x, y) \in (P, \bar{P})} f(x, y) = f(P, \bar{P}) \\ &\leq \sum_{(x, y) \in (P, \bar{P})} c(x, y) = c(P, \bar{P}), \quad \because \forall x, y \in V(D), f(x, y) \leq c(x, y) \\ &\leq \min c(P, \bar{P}) \end{aligned}$$

□

Corollary 26.4.2. *In a network N flow is the net flow into the sink of N .*

Synopsis. *Let $\bar{P} = \{t\}$, then by theorem $f(N)$ is the net flow into the sink.*

Proof. Suppose $P = V(D) - \{t\}$. Then by theorem, we have

$$\begin{aligned} f(N) &= \sum_{(x, y) \in (P, \bar{P})} f(x, y) - \sum_{(y, x) \in (\bar{P}, P)} f(y, x) \\ &= \sum_{x \in N^-(t)} f(x, t) - \sum_{x \in N^+(t)} f(t, x) \end{aligned}$$

□

Remark. *Exercise 5.1*

4. *Let N be a network with underlying digraph D which has a vertex $v \in V(D) - \{s, t\}$ with zero indegree. Clearly the flow into v is zero. Thus flow out of v is also zero by flow conservation equation. Let N' be the network obtained from N by deleting the vertex v . Then $f(N) = f(N')$.*

26.2 The Max-Flow Min-Cut Theorem

maximum flow A flow f in network N is maximum flow in N , if $f(N) \geq f'(N)$ for each flow f' in N .

minimum cut A cut (P, \bar{P}) in network N is minimum cut of N , if $c(P, \bar{P}) \leq c(X, \bar{X})$ for each cut (X, \bar{X}) in N .

f -unsaturated Let f be a flow in network N with underlying digraph D , and $Q = u_0, a_1, u_1, a_2, \dots, u_{n-1}, a_n, u_n$ be a semipath in D such that every forward arc $a_i = (u_{i-1}, u_i)$ has flow not upto its capacity, $f(a_i) < c(a_i)$ and every reverse arc $a_i = (u_i, u_{i-1})$ has some positive flow in it, $f(a_i) > 0$

f -augmenting semipath Let f be a flow in a network N with underlying digraph D . Suppose semipath $Q = s, a_1, u_1, a_2, \dots, u_{n-1}, a_n, t$ (from source to sink) is f -unsaturated, then Q is an f -augmenting semipath.

Theorem 26.5. Let f be a flow in a network N with underlying digraph D . The flow f is maximum in N iff there is no f -augmenting semipath in D .

Synopsis. Suppose Q is an f -augmenting semipath in D , then there exists a flow f^* in N such that $f(N) + \Delta = f^*(N)$. Therefore, f is not a maximum flow in N . Suppose there is no f -augmenting semipath in D , then there exists a cut (P, \bar{P}) such that $f(a) = c(a) \forall a \in (P, \bar{P})$ and $f(a) = 0 \forall a \in (\bar{P}, P)$. Suppose f^* in a maximum flow in N , then $f(N) \leq f^*(N) \leq c(P, \bar{P}) = f(N)$.

Proof. Let f be a flow in a network N with underlying digraph D and $Q = s, a_1, u_1, a_2, u_2, \dots, u_{n-1}, a_n, t$ be an f -augmenting semipath in D .

$$\text{define } \Delta_i = \begin{cases} c(a_i) - f(a_i) & \text{for every forward arc } a_i \in Q, \\ f(a_i) & \text{for every reverse arc } a_i \in Q, \end{cases}$$

Define $\Delta = \min\{\Delta_i\}$. Also define $f^* : E(D) \rightarrow \mathbb{Z}$ such that

$$f^*(a_i) = \begin{cases} f(a_i) + \Delta, & \text{for every forward arc } a_i \in Q, \\ f(a_i) - \Delta, & \text{for every reverse arc } a_i \in Q, \\ f(a_i), & \text{for every arc of } D \text{ which are not in } Q. \end{cases}$$

Since Q is an f -augmenting semipath in D , $\Delta > 0$ and $f(N) + \Delta = f^*(N)$.

Clearly $f(N) < f^*(N)$, and it is enough to show that f^* is a flow in N . f^* is a flow if it satisfies 1. capacity constraint and 2. conservation equation. For any arc $a_i \notin Q$, $f^*(a_i) = f(a_i) \leq c(a_i)$. Suppose $a_i \in Q$. If $a_i = (u_{i-1}, u_i)$, a_i is a forward arc and we have $f^*(a_i) = f(a_i) + \Delta \leq f(a_i) + \Delta_i = f(a_i) + c(a_i) - f(a_i) = c(a_i)$. If $a_i = (u_i, u_{i-1})$, then a_i is a reverse arc and we have $f^*(a_i) = \Delta \leq \min\{\Delta_i\} = \Delta_i = c(a_i)$. Thus f^* satisfies capacity constraint on every arc of D .

Let $x \in V(D) - \{s, t\}$. Suppose $x \notin Q$,

$$\begin{aligned} \text{Net flow out of } x &= \sum_{y \in N^+(x)} f^*(x, y) - \sum_{y \in N^-(x)} f^*(y, x) \\ &= \sum_{y \in N^+(x)} f(x, y) - \sum_{y \in N^-(x)} f(y, x) \\ &= 0 \end{aligned}$$

Suppose $x = u_i \in Q$, then Q has two arc having vertex x say, a_{i-1} , and a_i . There are four possibilities for these two arcs,

1. Both a_{i-1} , a_i are forward arcs.
2. Arc a_{i-1} is forward, but arc a_i is reverse.
3. Arc a_{i-1} is reverse, but arc a_i is forward.
4. Both a_{i-1} , a_i are reverse arcs.

Case 1 $a_{i-1} = (u_{i-1}, u_i)$ and $a_i = (u_i, u_{i+1})$.

$$\begin{aligned} \text{Net flow out of } x &= \sum_{y \in N^+(x)} f^*(x, y) - \sum_{y \in N^-(x)} f^*(y, x) \\ &= \sum_{\substack{y \in N^+(x) \\ y \neq u_{i+1}}} f^*(x, y) + f^*(u_i, u_{i+1}) - \left(\sum_{\substack{y \in N^-(x) \\ y \neq u_{i-1}}} f^*(y, x) + f^*(u_{i-1}, u_i) \right) \\ &= \sum_{\substack{y \in N^+(x) \\ y \neq u_{i+1}}} f(x, y) + f(u_i, u_{i+1}) + \Delta - \left(\sum_{\substack{y \in N^-(x) \\ y \neq u_{i-1}}} f(y, x) + f(u_{i-1}, u_i) \right) - \Delta \\ &= \sum_{y \in N^+(x)} f(x, y) - \sum_{y \in N^-(x)} f(y, x) \\ &= 0 \end{aligned}$$

Case 2 $a_{i-1} = (u_{i-1}, u_i)$ and $a_i = (u_{i+1}, u_i)$.

$$\begin{aligned} \text{Net flow out of } x &= \sum_{y \in N^+(x)} f^*(x, y) - \sum_{y \in N^-(x)} f^*(y, x) \\ &= \sum_{\substack{y \in N^+(x) \\ y \neq u_{i+1}, u_{i-1}}} f^*(x, y) + f^*(u_i, u_{i+1}) + f^*(u_i, u_{i-1}) - \sum_{y \in N^-(x)} f^*(y, x) \\ &= \sum_{\substack{y \in N^+(x) \\ y \neq u_{i+1}, u_{i-1}}} f(x, y) + f(u_i, u_{i+1}) + \Delta + f(u_i, u_{i-1}) - \Delta - \sum_{y \in N^-(x)} f(y, x) \\ &= \sum_{y \in N^+(x)} f(x, y) - \sum_{y \in N^-(x)} f(y, x) \\ &= 0 \end{aligned}$$

Case 3 $a_{i-1} = (u_i, u_{i-1})$ and $a_i = (u_i, u_{i+1})$.

$$\begin{aligned}
\text{Net flow out of } x &= \sum_{y \in N^+(x)} f^*(x, y) - \sum_{y \in N^-(x)} f^*(y, x) \\
&= \sum_{y \in N^+(x)} f^*(x, y) - \left(\sum_{\substack{y \in N^-(x) \\ y \neq u_{i-1}, u_{i+1}}} f^*(y, x) + f^*(u_{i-1}, u_i) + f^*(u_{i+1}, u_i) \right) \\
&= \sum_{y \in N^+(x)} f^*(x, y) - \left(\sum_{\substack{y \in N^-(x) \\ y \neq u_{i-1}, u_{i+1}}} f(y, x) + f(u_{i-1}, u_i) + \Delta + f(u_{i+1}, u_i) - \Delta \right) \\
&= \sum_{y \in N^+(x)} f(x, y) - \sum_{y \in N^-(x)} f(y, x) \\
&= 0
\end{aligned}$$

Case 4 $a_{i-1} = (u_i, u_{i-1})$ and $a_i = (u_{i+1}, u_i)$.

$$\begin{aligned}
\text{Net flow out of } x &= \sum_{y \in N^+(x)} f^*(x, y) - \sum_{y \in N^-(x)} f^*(y, x) \\
&= \sum_{\substack{y \in N^+(x) \\ y \neq u_{i-1}}} f^*(x, y) + f^*(u_i, u_{i-1}) - \left(\sum_{\substack{y \in N^-(x) \\ y \neq u_{i+1}}} f^*(y, x) + f^*(u_{i+1}, u_i) \right) \\
&= \sum_{\substack{y \in N^+(x) \\ y \neq u_{i-1}}} f(x, y) + f(u_i, u_{i-1}) - \Delta - \left(\sum_{\substack{y \in N^-(x) \\ y \neq u_{i+1}}} f(y, x) + f(u_{i+1}, u_i) \right) + \Delta \\
&= \sum_{y \in N^+(x)} f(x, y) - \sum_{y \in N^-(x)} f(y, x) \\
&= 0
\end{aligned}$$

Therefore, f^* is a flow on N . We have $f(N) < f^*(N)$. Thus f is not maximum flow in N due to the existence of an f -augmenting semipath in D .

Conversely, assume that there is no f -augmenting semipath in D . Now, we construct a cut (P, \bar{P}) of N . Let P be the set of all vertices $x \in V(D)$ such that there is an f -unsaturated $s - x$ semipath in D . Trivially, $s \in P$. And $t \notin P$ since there are no f -augmenting semipath in D .² Clearly, (P, \bar{P}) is a cut of the network N .

We claim that $c(P, \bar{P}) = f(N)$. Suppose there is a forward arc $(x, y) \in (P, \bar{P})$, then flow in it is saturated. If $f(x, y) < c(x, y)$, then there is an f -unsaturated $s - y$ semipath in D . ie, $s - x$ semipath + arc (x, y) . Thus every forward arc $(x, y) \in (P, \bar{P})$ is saturated. Suppose there is a reverse arc $(y, x) \in$

²An f -augmenting semipath is an f -unsaturated $s - t$ semipath in D .

(\bar{P}, P) , then there is no flow in it (saturated reversed arc). If $f(y, x) > 0$, then there is an f -unsaturated $s - y$ semipath in D . ie, $s - x$ semipath + arc (y, x) . Thus every reverse arc $(y, x) \in (\bar{P}, P)$ is saturated. And we have,

$$\begin{aligned} \sum_{(x,y) \in (P, \bar{P})} f(x, y) &= \sum_{(x,y) \in (P, \bar{P})} c(x, y) \\ \sum_{(y,x) \in (\bar{P}, P)} f(y, x) &= 0 \\ f(N) &= \sum_{(x,y) \in (P, \bar{P})} f(x, y) - \sum_{(y,x) \in (\bar{P}, P)} f(y, x) \\ &= \sum_{(x,y) \in (P, \bar{P})} c(x, y) \\ &= c(P, \bar{P}) \end{aligned}$$

Suppose f^* is maximum flow in network N and (X, \bar{X}) is minimum cut of N . Then $f(N) \leq f^*(N)$. Thus we have, $f(N) \leq f^*(N) \leq c(X, \bar{X}) \leq c(P, \bar{P}) = f(N)$. Therefore, $f(N) = f^*(N)$. ie, the flow f is maximum in network N if there are no f -augmenting semipaths in D . \square

Theorem 26.6 (maximum-flow, min-cut). *In every network, the value of maximum flow equals capacity of minimum cut.*

Proof. Suppose flow f in network N is maximum, then by previous theorem there is no f -augmenting semipath in D . And $f(N) \leq c(X, \bar{X})$ for any cut (X, \bar{X}) in N . We can construct a cut (P, \bar{P}) in N such that $f(N) = c(P, \bar{P})$. Let P be the set of all vertices x in D such that there is an f -unsaturated $s - x$ semipath in D . Clearly $s \in P$ and $t \notin P$. Also $f(P, \bar{P}) = c(P, \bar{P})$ and $f(\bar{P}, P) = 0$. Then the cut (P, \bar{P}) is minimum cut of N . Suppose there is a cut (X, \bar{X}) such that $c(X, \bar{X}) < c(P, \bar{P})$. Then $f(N) = f(P, \bar{P}) - f(\bar{P}, P) = c(P, \bar{P}) < c(X, \bar{X})$ which is a contradiction. Therefore, the value of maximum flow equals capacity of minimum cut. \square

Remark. Exercise 5.2

1. Suppose (X, \bar{X}) is a cut of N such that $f(a) = c(a)$, $\forall a \in (X, \bar{X})$ and $f(a) = 0$, $\forall a \in (\bar{X}, X)$. By the definition of cut, $s \in X$ and $t \in \bar{X}$. Thus there is no f -augmenting semipath in D . Suppose there is an f -augmenting semipath Q in D , then there is either (a) a forward arc $(x, y) \in (X, \bar{X})$ such that $f(x, y) < c(x, y)$ or (b) a reverse arc $(y, x) \in (\bar{X}, X)$ such that $f(y, x) > 0$ which is a contradiction. Therefore, the flow $f(N)$ is maximum and the given cut (X, \bar{X}) is minimum as shown in the proof of the maximum-flow min-cut theorem.
3. The algorithm suggested in the hint of this exercise won't work if two subnetworks have a common arc such that the direction of flow in which is not consistent. Suppose, the generalized network is not supposed to have any common arcs. Then construct subnetworks for each pair (s, t) with all those arcs which are on some $s - t$ semipath. Define subnetwork capacity function $c'(a) = c(a)$ for every arc in N' .

Let N be a generalized network with set of sources S and set of sinks T . A flow in N is maximum if there is not f -augmenting $s - t$ semipath for each pair $(s, t) \in S \times T$.

26.3 A max-flow min-cut algorithm

Theorem 26.7. Let N be a network with underlying digraph D , source s , sink t , capacity function c and flow f . Let D' be the digraph with same vertex set as D and arc set defined by $E(D') = \{(x, y) : (x, y) \in E(D), c(x, y) > f(x, y) \text{ or } (y, x) \in E(D), f(y, x) > 0\}$. ie, D' has only the unsaturated arcs of D . Then D' has an $s - t$ directed path iff D has an f -augmenting semipath. Moreover, shortest $s - t$ path in D' has the same length as shortest f -augmenting semipath in D .

Synopsis. Each directed $s - t$ path in D' has respective f -augmenting semipath in D and vice versa. Clearly, they have the same length.

Proof. Let N be a network with underlying digraph D , capacity c and flow f . Let D' be the digraph with vertex set $V(D') = V(D)$ and arc set $E(D') = \{(x, y) : \text{either } (x, y) \text{ or } (y, x) \text{ is unsaturated in } N\}$.

Suppose D' has a directed $s - t$ path $Q' : s, u_1, u_2, \dots, u_{n-1}, t$. Then by the construction of D' , for each $u_i \in Q$, there exists an f unsaturated arc a_i in D . ie, either forward arc $a_i = (u_{k-1}, u_k)$ such that $f(u_{k-1}, u_k) < c(u_{k-1}, u_k)$ or reverse arc $a_i = (u_k, u_{k-1})$ such that $f(u_k, u_{k-1}) > 0$. Therefore, we have an $s - t$ semipath $Q : s, a_1, u_1, a_2, \dots, u_{n-1}, a_n, t$ in D such that Q is an f -augmenting semipath since every arc in Q is f -unsaturated. Clearly, Q, Q' are of the same length.

Conversely, suppose that the digraph D has an f -augmenting semipath $Q : s, a_1, u_1, a_2, \dots, u_{n-1}, a_n, t$. Then each arc $a_i \in Q$ are f -unsaturated and by the construction of D' , there exists a directed $s - t$ path $Q' = s, u_1, u_2, \dots, u_{n-1}, t$ in D' . And Q, Q' are of the same length.

There is a one-one correspondence between the directed $s - t$ paths in D' and f -augmenting semipaths in D . Clearly, they have the same length. Thus shortest directed $s - t$ path in D and shortest f -augmenting semipath in D' are of the same length. \square

saturation arc of N with respect to the flow f is an arc a_j in an f -augmenting semipath Q with $\Delta_j = \Delta$.

augmentation path is an f -augmenting semipath Q in D .

Algorithm 26.8 (max-flow min-cut). An algorithm to find maximum flow and minimum cut of a network N with underlying digraph D , source s , sink t , capacity function c and initial flow f .

1. Construct digraph D' with vertex set $V(D') = V(D)$ and arc set $E(D') = \{(x, y) : (x, y) \in E(D) \& f(x, y) < c(x, y) \text{ or } (y, x) \in E(D) \& f(y, x) > 0\}$

2. Find (shortest) $s-t$ directed path in D' using Moore's breadth first search(BFS) algorithm. If D' doesn't have an $s-t$ path, then proceed to step 5. Otherwise, let $Q' : s, u_1, u_2, \dots, u_{n-1}, t$ be a (shortest) $s-t$ path in D' .
3. Let $Q : s, a_1, u_1, a_2, \dots, u_{n-1}, a_n, t$ be the respective semipath in D such that $f(a_j) < c(a_j)$ for forward arcs and $f(a_i) > 0$ for reverse arcs. Let $\Delta_j = c(a_j) - f(a_j)$ for forward arcs and $\Delta_j = f(a_j)$ for reverse arcs. And let $\Delta = \min\{\Delta_j\}$. And augment flow f by Δ ie, $f(a_j) \leftarrow f(a_j) + \Delta$ for forward arcs and $f(a_j) \leftarrow f(a_j) - \Delta$ for reverse arcs.
4. Goto step 1 (Proceed with new flow f and find whether there are any directed $s-t$ paths in D' . If any, augment the flow along the new augmentation path Q by saturating the flow along the saturation arc.)
5. There is no $s-t$ directed path in D' . Thus there is no f -augmenting semipath in D . Therefore the flow f in N is maximum. Let P be the set of all vertices in D' with non-zero breadth first index(bfi) from Moore's BFS algorithm applied in step 2. (P, \bar{P}) is minimum cut of N .

Remark. Validity of the algorithm is proved in the previous theorem.

26.5 Connectivity and Edge-Connectivity

edge cutset is the set U subset of $E(G)$ such that $G - U$ is disconnected.

vertex cutset is the set S subset of $V(G)$ such that $G - S$ is disconnected.

edge connectivity $\lambda(G)$ is the minimum cardinality of all edge cutsets of G .

connectivity $\kappa(G)$ is the minimum cardinality of all vertex cutsets of G .

Theorem 26.9. For every graph G , $\kappa(G) \leq \lambda(G) \leq \delta(G)$

Proof. Suppose graph G is disconnected then $\kappa(G) = \lambda(G) = \delta(G) = 0$. Let G be a connected graph. Then G has at least one vertex v with degree $\delta(G)$. Therefore $\lambda(G) \leq \delta(G)$ since edges incident with v form an edge cutset of G and $\lambda(G)$ is the cardinality of all edge cutsets.

Let G be a graph with edge connectivity $\lambda(G) = c$. Let U be a edge cutset with cardinality c and let edge $uv \in U$. Construct a set of vertices $S \subset V(G)$ such that (S is of minimal cardinality and) for each edge in U other uv , S has a vertex incident with it. Cardinality of S is atmost $c - 1$, since we can select one vertex each for each edge in U other than u, v . If $G - S$ is a disconnected graph, then $\kappa(G) < \lambda(G)$. Suppose $G - S$ is a connected graph, then delete a non-pendent vertex u or v from $G - S$, say v . Since $G - S$ is a connected graph with a singleton edge cutset, $\{uv\}$. We have a vertex cutset $S \cup \{v\}$ of G . Therefore, $\kappa(G) \leq c = \lambda(G)$. \square

Theorem 26.10. If G is a graph of diameter 2, then $\lambda(G) = \delta(G)$

n -edge connected G is n -edge connected if $\lambda(G) \geq n$.

n connected G is n -connected if $\kappa(G) \geq n$.

Theorem 26.11. *Let G be a graph of order p and n be an integer such that $1 \leq n \leq p-1$. If $\delta(G) \geq \frac{p+n-2}{2}$, then G is n -connected.*

connection number $c(G)$ is the smallest integer such that $2 \leq c(G) \leq p$ and every subgraph of order n in G is connected.

l -connectivity $\kappa_l(G)$ is minimum number of vertices whose removal will produce a disconnected graph with at least l components or a graph with fewer than l vertices.

(n, l) -connected A graph G is (n, l) -connected if $\kappa_l(G) \geq n$.

Remark. *Exercises 5.5*

1. $\lambda(K_{m,n}) = \kappa(K_{m,n}) = m$

8. $c(K_p) = 2$, $c(K_{m,n}) = n+1$, $c(C_p) = p-1$

Every two vertices of complete graph of order p are adjacent. For complete bi-partite graph $K_{m,n}$ such that $1 \leq m \leq n$, there exists a totally disconnected subgraph of order n . Therefore $c(K_{m,n}) \geq n+1$. And with $n+1$ vertices, both partitions have at least two vertices each and therefore the graph is connected and $c(K_{m,n}) \leq n+1$. For cycle C_p , any subgraph is disconnected if two non-adjacent vertices are deleted. Therefore $c(C_p) \geq p-1$. And C_p remains connected even after deletion of any vertex, therefore $c(C_p) \leq p-1$.

- 9.

$$\delta(G) \geq \frac{p + (l-1)(n-2)}{l} \implies \kappa_l(G) \geq n$$

26.6 Menger's Theorem

Theorem 26.12. *For a non-trivial graph G , $\lambda(u, v) = M'(u, v)$ for every pair (u, v) of vertices of G .*

Corollary 26.12.1. *Graph G is n -edge connected iff every two vertices of G are connected by at least n edge disjoint paths.*

Theorem 26.13. *For every pair of non-adjacent vertices u, v in graph G , $\kappa(u, v) = M(u, v)$.*

Corollary 26.13.1. *Graph G is n -connected iff every pair of vertices of G are connected by at least n internally disjoint paths.*

Algorithm 26.14 (connectivity $\kappa(G)$). .

1. If degree of every vertex is $p-1$, then output $\kappa = p-1$ and stop. Otherwise, continue.
2. If G is disconnected, output $\kappa = 0$ and stop. Otherwise, continue.
3. $\kappa \leftarrow p$
4. $i \leftarrow 0$

5. If $i \leq \kappa$, then $i \leftarrow i + 1$ and continue. Otherwise, output κ and stop.
6. $j \leftarrow i + 1$
7. (1) If $j = p + 1$, then return to step 5. Otherwise continue.
 - (2) If $v_i v_j \notin E(G)$, construct network N with digraph D as follows :
 for each vertex $v \in V(G)$, there are two vertices $v', v'' \in V(D)$ and
 an arc $(v', v'') \in E(D)$. And for each edge $uv \in E(G)$, there are
 two arcs $(u'', v), (v'', u) \in E(D)$. The capacity function is given by,
 $c(v', v'') = 1$ for every $v \in V(G)$ and $c(a) = \infty$ for every other arc
 in D . Set source $s = v''_i$ and sink $t = v'_j$ and find maximum flow in
 N using max-flow min-cut algorithm. Otherwise proceed to step 7d
 - (3) If $f(N) < \kappa$, then $\kappa \leftarrow f(N)$. Otherwise, continue.
 - (4) $j \leftarrow j + 1$ and return to step 7a

Chapter 27

Matchings and Factorizations

27.1 An Introduction to Matching

Marriage Problem Given a collection of men and women, where each woman knows some of the men. Can every woman marry a man she knows ?

Assignment Problem Given several job openings and applicants for one or more of these positions. Find an assignment so that maximum positions are filled ?

Optimal Assignment Problem Given several job openings and applicants for one or more of these positions. The benefits of employing these applicants on those positions are also given. Find an assignment of maximum benefit to the company ?

matching in G is a 1-regular¹ subgraph of G .

maximum matching in G is a matching of G with maximum cardinality.

perfect matching in G is a matching of cardinality $p/2$. ie, $p/2$ edges.

maximum weight matching in a weighted graph G is a matching with maximum weight.

Definitions 27.1. *Let M be a matching in a graph G ,*

matched edge *is an edge in subgraph M of G .*

unmatched edge *is an edge of G that doesn't belong to M .*

matched vertex *with respect to M is a vertex incident with an edge of M .*

single vertex *is a vertex that is not incident with any edge of M .*

alternating path *in G is a path with edges alternately matched and unmatched.*

¹A graph G is k -regular, if every vertex of G has degree k .

augmenting path in G is a non-trivial alternating path with single vertices as end vertices.

Theorem 27.2. Let M_1, M_2 be two matchings in G such that there is a spanning subgraph H of G with edges that are either in M_1 or M_2 , but not both. Then the components of H are either 1. isolated vertex 2. even cycle with edge alternately from M_1 and M_2 3. a non-trivial path with edges alternately from M_1 and M_2 such that each end vertex is single with respect to either M_1 or M_2 , but not both.

Synopsis. $\Delta(H) \leq 2$ by Pigeonhole principle. Any component of H is either a path or a cycle. A cycle with edge alternately from M_1 and M_2 is even. If an end vertex of a non-trivial path is matched with respect to M_1 (WLOG), then it is there in $M_1 - M_2$ ie, it is not there in M_2 . If there is another edge in M_2 incident with it, then it has to be in H and it will cease to be an end vertex of path component. Therefore, it is unmatched with respect to M_2 .

Theorem 27.3. A matching M in a graph G is maximum iff there is no augmenting path with respect to M in G .

Synopsis. If M is maximum matching and P an M -augmenting path. Since both end-vertices are single, length of P is odd. Let M', M'' be edges of P which are in M and not in M respectively. Then $M - M' + M''$ is a matching of cardinality one greater than that of M which is a contradiction since M is maximum. Conversely, suppose M be a matching such that there no M -augmenting paths in G . Let M' be a maximum matching in G . Then a nontrivial path component of the graph induced by $M \Delta M'$ is of even length otherwise both end-vertices are matched with respect to one of the matching M or M' which is a contradiction. Again every cycle components are even. Therefore $|M| = |M'|$, since $M \Delta M'$ doesn't have a nontrivial component of another kind.

Definitions 27.4. Let U_1, U_2 be two nonempty, disjoint, subsets of the vertex set of a graph G . Then U_1 **is matched to** U_2 if there exists a matching M in G such that every edge in M incident with a vertex in U_1 and a vertex in U_2 . And every vertex of U_1 (or U_2) is incident with some edge in M . Suppose M^* be a matching such that $M \subset M^*$, then U_1 **is matched under M^* to** U_2 .

Definitions 27.5. Let U be a nonempty set of vertices of a graph G . U is **nondeficient**,² if $|N(S)| \geq |S|$ for every nonempty subset S of U .

Theorem 27.6. Let G be a bipartite graph with partite sets V_1, V_2 . The set V_1 can be matched to a subset of V_2 iff V_1 is nondeficient.

Corollary 27.6.1. Every r -regular bipartite multigraph has a perfect matching.

Theorem 27.7. A collection S_1, S_2, \dots, S_n of finite non-empty sets has a system of distinct representatives iff for each k , $0 \leq k \leq n$, the union of any k of these sets contains at least k elements.

Remark (Hall's Marriage Theorem). Suppose there are n women. Then every women can marry a man she knows iff each subset of k women ($1 \leq k \leq n$) collectively knows atleast k men.

Remark. Let W be a set of n women. Then there are $2^n - 1$ nonempty subset for W . Thus, Hall's Marriage Theorem suggests that we ensure $|N(S)| \geq |S|$ for every nonempty subset S of W . This method has complexity $O(2^n)$.

² $N(S)$ is the neighbourhood set of all vertices adjacent to some vertex in S

27.2 Maximum Matching in Bipartite Graphs

Definitions 27.8. Let M be a matching in a graph G and P is an augmenting path with respect to M . Let M' be set of edge in P and M . And M'' be the set of edges in P and not in M . Then $M_1 = (M - M') \cup M''$ is the **matching obtained by augmenting M along path P** .

Remark. $|M_1| = |M| + 1$

Theorem 27.9. Let M be a a matching of a graph G that is not maximum, and let v be a single vertex with respect to M . Let M_1 denote the mathing obtained by augmenting M along some augmenting path. If G contains an augmenting path with respect to M_1 that has v and an end-vertex, then G contains an augmenting path with respect to M that has v as an end-vertex

Corollary 27.9.1. Let M be a matching of a graph G . Suppose that $M = M_1, M_2, \dots, M_k$ is a finite sequence of matchings of G such that M_i ($2 \leq i \leq k$) is obtained by augmenting M_{i-1} along some augmenting path. Suppose v is a single vertex with respect to M for which there exists no augmenting path starting at v . Then G does not contain an augmenting path with respect to M_i ($2 \leq i \leq k$) that has v as an end-vertex.

Definitions 27.10. An **alternating tree** with respect to a matching M is a tree such that every path from it's root are alternating path with respect to M .

Algorithm 27.11 (Maximum Matching Algorithm for Bipartite Graphs). .

1. $i \leftarrow 1$ and $M \leftarrow M_1$
2. If $i < p$, then continue; otherwise stop.
3. If v_i is matched, then $i \leftarrow i + 1$ and return to Step 2;
otherwise, $v \leftarrow v_i$ and Q is initialized to contain v only.
4. (1) For $j = 1, 2, \dots, p$ and $j \neq i$, let $TREE(v_j) \leftarrow F$.
Also, $TREE(v_i) \leftarrow T$.
(2) If $Q = \phi$, then $i \leftarrow i + 1$ and return to Step 2;
otherwise, delete a vertex x from Q and continue.
(3) (1) Suppose that $N(x) = \{y_1, y_2, \dots, y_k\}$. Let $j \leftarrow 1$.
(2) If $j \leq k$, then $y \leftarrow y_j$; otherwise return to Step 4.2
(3) If $TREE(y) = T$, then $j \leftarrow j + 1$ and return to Step 4.3.2.
Otherwise, continue.
(4) If y is incident with a matching edge yz , then $TREE(y) \leftarrow T$,
 $TREE(z) \leftarrow T$, $PARENT(y) \leftarrow x$, $PARENT(z) \leftarrow y$ and add
 z to Q , $j \leftarrow j + 1$ and return to Step 4.3.2. Otherwise, y is a
single vertex and continue.
(5) Use $PARENT$ to determine the alternating $v - x$ path P' in the
alternating tree. Let P be the augmenting path obtained from P'
by adding the path x, y . Proceed to Step 5
5. Augment M along P to obtain a new matching M' . Let $M \leftarrow M'$, $i \leftarrow i + 1$, and return to Step 2.

Definitions 27.12. Let G be a weighted complete bipartite graph with partite sets V_1 and V_2 . A **feasible vertex labeling** is a real function $l : V(G) \rightarrow \mathbb{R}$ on vertex set of G such that $l(v) + l(u) \geq w(vu)$ where $v \in V_1$ and $u \in V_2$.

Definitions 27.13. Consider the function $l : V(G) \rightarrow \mathbb{R}$ such that $\forall v \in V_1, l(v) = \max\{w(vu) : u \in V_2\}$ and $\forall u \in V_2, l(u) = 0$. Then l is a feasible vertex labeling on $V(G)$. And,

E_l is the set of all edge of the weighted complete bipartite graph G such that $l(v) + l(u) = w(vu)$.

H_l is the spanning subgraph of G induced by the edge set E_l .

Theorem 27.14. Let l be a feasible vertex labeling of a weighted complete bipartite graph G . If H_l contains a perfect matching M' , then M' is a maximum weight matching of G .

Algorithm 27.15 (Kuhn-Munkres). .

1. (1) For each $v \in V_1$, let $l(v) \leftarrow \max\{w(vu) : u \in V_2\}$.
 (2) For each $u \in V_2$, let $l(u) \leftarrow 0$.
 (3) Let H_l be the spanning subgraph of G with edge set E_l .
 (4) Let G_l be the underlying graph of H_l .
2. Apply Algorithm ?? to determine a maximum matching M in G_l .
3. (1) If every vertex v of V_1 is matching with respect to M , output M and stop. Otherwise, continue.
 (2) Let x be the first single vertex of V_1 .
 (3) Construct an alternating tree with respect to M that is rooted at x . If an augmenting path P is discovered, then augmenting M along P and return to Step 3.1. Otherwise, let T be the alternating tree with respect to M and rooted at x that cannot be expanded further in G_l .
4. Compute $m_l \leftarrow \min\{l(v) + l(u) - w(vu) : v \in V_1 \cap V(T), u \in V_2 - V(T)\}$.
 Let

$$l'(v) = \begin{cases} l(v) - m_l & \text{for } v \in V_1 \cap V(T) \\ l(v) + m_l & \text{for } v \in V_2 \cap V(T) \\ l(v) & \text{otherwise} \end{cases}$$

5. Let $l \leftarrow l'$, construct G_l and return to Step 3.3.

27.4 Factorizations

Definitions 27.16. A **factor** of a graph G is a spanning³ subgraph of G .

Definitions 27.17. Let G_1, G_2, \dots, G_n be edge-disjoint factors of G such that $E(G) = \cup_{i=1}^n E(G_i)$. Then G is **factorable** and $G = G_1 \oplus G_2 \oplus \dots \oplus G_n$.

³Spanning subgraph of a graph G has every vertex of G

Definitions 27.18. An r -regular factor of G is an **r -factor** of G .

Definitions 27.19. If G has a factorisation to r -factors, then G is **r -factorable**.

Remark. $K_{3,3}$ is 1-factorable. K_5 is 2-factorable.

Definitions 27.20. An **odd component of G** is a component of G with odd number of vertices. And an **even component of G** is a component of G of with even number of vertices.

Theorem 27.21 (Tutte). A nontrivial graph G has a 1-factor iff for every proper subset S of $V(G)$, the number of odd components of $G - S$ does not exceed $|S|$.

Remark. There exist cubic graphs that doesn't have a 1-factor.

Theorem 27.22 (Petersen). Every bridgeless cubic graph contains a 1-factor.

Remark. Every bridgeless cubic graphs has a 1-factor. Let G be a bridgeless cubic graph. Consider every pair of factors G_1, G_2 such that $G = G_1 \oplus G_2$ where G_1 is a 1-factor and G_2 is a 2-factor. G is not 1-factorable only if every such G_2 doesn't have a 1-factor.

Theorem 27.23. Petersen graph is not 1-factorable.

Theorem 27.24. Every r -regular bipartite multigraph ($r \geq 1$) is 1-factorable.

Remark (Application of 1-factorisation). For even number p , a 1-factorisation of K_p corresponds to the schedule of a round of the round robin tournament among p teams. If p is odd, consider K_{p+1} where v_{p+1} is an imaginary team called bye team. A game with bye team is a bye.

Definitions 27.25. A **hamiltonian cycle** is a spanning cycle. And, **Hamiltonian graph** is a graph containing a hamiltonian cycle.

Theorem 27.26. Complete graph K_{2n+1} can be factored into n hamiltonian cycles.

Remark. For $n = 3$, K_7 can be factored into three hamiltonian cycles.

Theorem 27.27. Let $0 \leq r < p$. Then there exists an r -regular graph of order p iff pr is even.

Definitions 27.28. Let $\{E_1, E_2, \dots, E_n\}$ be partition of $E(G)$. And let H_i be subgraph of G induced by the edge set E_i . A **decomposition** of a graph G is a collection of these subgraphs H_1, H_2, \dots, H_n . And $G = H_1 \oplus H_2 \oplus \dots \oplus H_n$.

Definitions 27.29. Let $G = H_1 \oplus H_2 \oplus \dots \oplus H_n$ be a decomposition of G such that $H \cong H_i$. Then G is **H -decomposable**.

Remark. $K_{3,3}$ is $3K_2$ -decomposable. K_5 is C_5 -decomposable. K_{2n} is nK_2 -decomposable. K_{2n+1} is C_{2n+1} -decomposable.

Every graph is K_2 -decomposable. Every complete bipartite graph $K_{m,n}$ is $K_{1,m}$ -decomposable and $K_{1,n}$ -decomposable.

27.5 Block Designs

Definitions 27.30. A block design on a set V is a collection of k -element subsets of V such that each element of V appears exactly in r subsets.

variety The elements of V are called varieties.

block k -element subsets of V are called blocks.

balanced design If each variety appears in exactly r blocks and each pair of varieties appears in exactly λ blocks.

incomplete design If blocks are proper subsets of V . ie, $k < v$.

Definitions 27.31. A balanced incomplete block design of v varieties in b blocks of cardinality k such that each variety appears in exactly r blocks and each pair of varieties appears in exactly λ blocks is a (b, v, r, k, λ) -design.

Theorem 27.32. $bk = vr$

Theorem 27.33. $\lambda(v-1) = r(k-1)$

Corollary 27.33.1. $\lambda < r$

Theorem 27.34 (Fisher's Inequality). $b \geq v$

symmetric design If $b = v$

Theorem 27.35. In a symmetric (b, v, r, k, λ) -design with even v , $r - \lambda$ is a perfect square.

steiner triple system (b, v, r, k, λ) -design with $k = 3$, $\lambda = 1$.

Remark. (b, v, r, k, λ) -designs are incomplete. However, a complete block design (ie, $v = 3$) is also included as a Steiner triple system.

Theorem 27.36. Steiner triple system with v varieties exists iff $v = 6n + 1$ or $v = 6n + 3$ or $v = 3$.

Definitions 27.37 (Kirkman's Schoolgirls Problem). A class of 15 girls. Parade 15 girls in five rows (3 girls in a row). Is it possible to plan 7 days parade so that two girls are together in a row exactly once ?

kirkman triple system Steiner triple system with $v = 6n + 3$.

Remark. It is proved that kirkman triple system exists with $v = 6n + 3$ for every $n \geq 0$.

Theorem 27.38. The code consisting of the rows of the incidence matrix of a (b, v, r, k, λ) -design ($b = v$, $r=k$) is t -error correcting, where $t = k - \lambda - 1$.

Part XIX

ME800403 Combinatorics