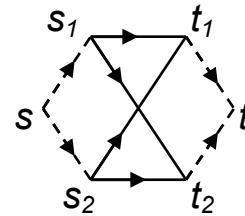


Networks and Flows (continued)

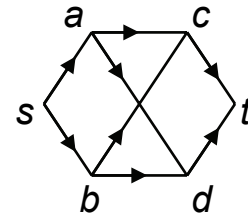
Generalizations

Results studied in the previous lecture, e.g., the *max-flow min-cut theorem*, apply to more general situations, after a reduction is done to the kind of a directed network that we considered earlier. Examples follow.



- A directed network with *multiple sources and/or sinks* can be handled as a network with one source and one sink, if you introduce two additional nodes s and t and connect them to each source and sink by arcs of sufficient capacity.
- In a *graph with undirected edges*, each undirected edge xy can be represented as two directed arcs, xy and yx .
- To *count edge-disjoint paths*, assign a unit capacity to each arc in the directed network. Then the max-flow min-cut theorem yields the following result (Menger, 1927):

- The maximum number of edge-disjoint paths between two vertices, s and t , in a directed graph is the minimum number of edges whose removal makes t unreachable from s .

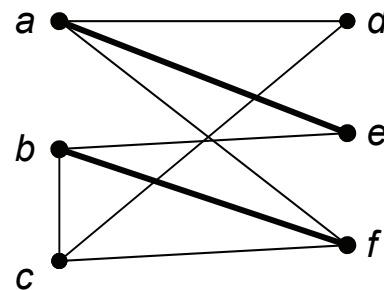


Another example is a *matching problem* that we consider next.

Matching Problem

Definitions:

- A **matching** in a graph is a set of edges that do not share vertices with each other.
- Vertices that are incident with matching edges are called **saturated**.
- A matching is **perfect** iff it saturates all vertices.
- A **maximal matching** saturates as many vertices as possible.



Therefore, a perfect matching partitions the set of vertices into two subsets of equal size and establishes a 1-1 correspondence.

Marriage Theorem (Hall, 1935):

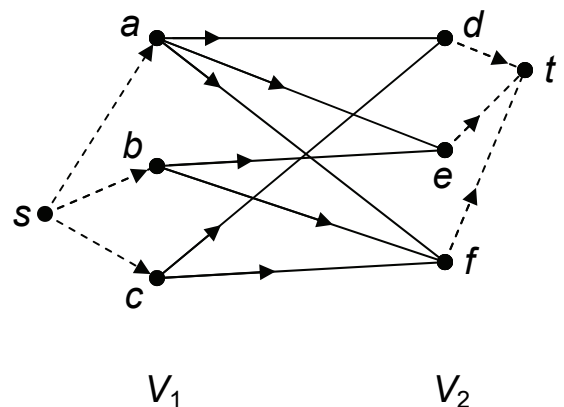
If G is a bipartite graph with bipartition sets V_1 and V_2 , then

$$\exists \text{ matching that saturates } V_1 \iff |X| \leq |A(X)| \quad \forall X \subseteq V_1, \quad (*)$$

where $A(X)$ is the subset of vertices in V_2 that are adjacent to X .

Proof: (\rightarrow) is straightforward, since every element of X is saturated.

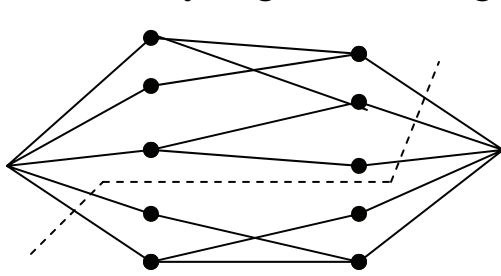
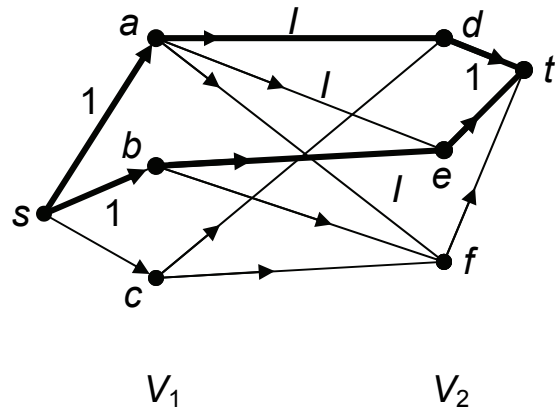
(\leftarrow) Assuming the r.h.s. of (*), add a source s connected to all V_1 and a sink t connected to all V_2 ; assign a unit weight and the direction from s to t to all new arcs; assign an int weight $I > |V_1|$ and the direction $V_1 \rightarrow V_2$ to all original arcs of G . Then there is a 1-1 correspondence between matchings of G and (s, t) – flows, so that in each case the value of the flow is equal to the number of edges in the matching.



a flow $F \leftrightarrow$ a matching

$$\text{val}(F) = |\text{edges of matching}|$$

Suppose that there is no perfect matching. Then every flow value is $< |V_1|$, therefore, by the max-flow min-cut theorem, there is an (S, T) – cut whose capacity is less than $|V_1|$. This cut cannot include any edge connecting V_1



and V_2 , since each of these edges has capacity $I > |V_1|$. Therefore, if X is the intersection of V_1 and S , then the (S, T) – cut must go through edges connecting s with $V_1 \setminus X$ and edges connecting $A(X)$ with t . Each

of these edges has the unit capacity.

Therefore, if

$$\text{cap}(S, T) < |V_1|, \rightarrow \text{cap}(S, T) = |V_1 \setminus X| + |A(X)| \geq |V_1|, \quad \perp.$$

$$X = V_1 \cap S$$

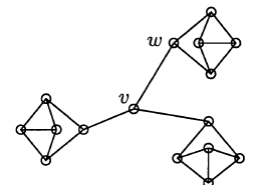
We have a contradiction. Therefore, a perfect matching exists.

Propositions for a graph G with a vertex set V :

- If G has a perfect matching, then $|V|$ is even.
- If G has a Hamiltonian path or cycle, then G has a perfect matching iff $|V|$ is even.
- If $|V|$ is even, and every vertex has degree $d \geq \frac{1}{2}|V|$, then G has a perfect matching.

Proof: by Dirac theorem, if $|V| \geq 4$, then G is Hamiltonian.

Exercises 10, 11 page 461: Suppose G is connected, $|V|$ is even, every vertex has degree 2 (3). Must G have a perfect matching? –Yes for degree 2. No for degree 3: Figure \rightarrow



The Integers (continued)

Congruence

Definitions. Let a , b and n be integers, with $n > 1$. Then:

We say that a is **congruent** to b **modulo** n , and write $a \equiv b \pmod{n}$, iff $n \mid (a - b)$, i.e., n divides $a - b$. In this case n is called the **modulus** of the congruence.

Congruence is an equivalence relation (it is *reflexive*, *symmetric*, *transitive*), and therefore it defines equivalence classes, called congruence classes, that partition the integers.

The **congruence class** mod n of a , written \bar{a} or $n\mathbb{Z} + a$, is

$$\bar{a} = \{b \in \mathbf{Z} \mid a \equiv b \pmod{n}\}.$$

Replacement of a by its remainder upon division by n , written as “ $a \pmod{n}$ ”, is called **reduction modulo n** .

Reductions of results of familiar binary operations: $a + b \pmod{n}$ and $ab \pmod{n}$, are called **addition modulo n** and **multiplication modulo n** .

The solution x , $0 \leq x < n$, of the congruence $a + x = 0 \pmod{n}$ is called **additive inverse** of a modulo n . It always exists: to find it, one can take the reduction modulo n of $(-a)$.

The solution to $ax = 1 \pmod{n}$, $0 \leq x < n$, is called **multiplicative inverse** modulo n , written $x = a^{-1}$.

Propositions: here, again, let a , b and n be integers, with $n > 1$.

- The following five statements are equivalent:

1. $n \mid (a - b)$

2. $a \equiv b \pmod{n}$

3. $a \in \bar{b}$

4. $b \in \bar{a}$

5. $\bar{a} = \bar{b}$

- $a \equiv b \pmod{n} \leftrightarrow \bar{a} = \bar{b}$

- There are n non-overlapping congruence classes of integers modulo n .

Proof: By the Division Algorithm theorem, every integer a has unique decomposition $a = qn + r$ with $0 \leq r < n$, and therefore belongs to exactly one of n congruence classes modulo n : to \bar{r} .

- If $a \equiv x \pmod{n}$ and $b \equiv y \pmod{n}$, then

- (a) $a + b \equiv x + y \pmod{n}$

- (b) $ab \equiv xy \pmod{n}$

Proof by direct application of the definition of congruence:

$$n \mid (a - x), \quad n \mid (b - y) \rightarrow$$

$$(a + b) - (x + y) = (a - x) + (b - y) \text{ is divisible by } n,$$

$$ab - xy = ab - ay + ay - xy = a(b - y) + (a - x)y \text{ is divisible by } n.$$

Therefore, we can add, subtract and multiply congruence equations, side-by-side. However, we must be careful when dividing them. In general, the congruence $ac \equiv bc \pmod{n}$ with some integer $c > 1$ does not imply the congruence $a \equiv b \pmod{n}$. Nevertheless, the following propositions hold:

- $ac \equiv bc \pmod{n}$, $\gcd(c, n) = 1 \rightarrow a \equiv b \pmod{n}$.

Proof: It follows from the Euclidean Algorithm that

$$\begin{aligned} \exists x, y \in \mathbf{N} \ni cx + ny &= \gcd(c, n) = 1, \\ \exists k \in \mathbf{Z} \ni ac - bc &= nk \rightarrow \\ (a - b)cx &= nkx \rightarrow \\ (a - b)(1 - ny) &= nkx \rightarrow \\ a - b &= n(kx + (a - b)y) \rightarrow \\ n \mid (a - b) &\rightarrow a \equiv b \pmod{n}. \end{aligned}$$

- $\gcd(a, n) = 1 \rightarrow a$ has a multiplicative inverse, and $\forall b \in \mathbf{Z} \exists ! \bar{x} \pmod{n} \ni ax \equiv b \pmod{n}$.

Simple Congruence Problems and Methods of Their Solution:

$$3x \equiv 1 \pmod{5} \Rightarrow x = 2 \text{ (try all } x \text{ from 0 to 4)}$$

$$3x \equiv 1 \pmod{6} \Rightarrow \perp \text{ (consider 2 cases: an odd } x \text{ and an even } x)$$

$$3x \equiv 3 \pmod{6} \Rightarrow x \in \{1, 3, 5\} \text{ (use the same strategy)}$$

$$\left\{ \begin{array}{l} 2x + 3y \equiv 1 \pmod{6} \\ x + 3y \equiv 4 \pmod{6} \end{array} \right\} \Rightarrow 3x + 6y \equiv 5 \pmod{6} \Rightarrow \perp \text{ (take the sum of two equations)}$$

$$\left\{ \begin{array}{l} 2x + 3y \equiv 1 \pmod{6} \\ x + 3y \equiv 5 \pmod{6} \end{array} \right\} \Rightarrow \left\{ \begin{array}{l} 3x \equiv 0 \rightarrow x \in \{\bar{0}, \bar{2}, \bar{4}\} \\ x + 3y \equiv 5 \pmod{6} \end{array} \right\} \Rightarrow x \equiv 2 \pmod{6}, y \equiv \begin{cases} 1 \pmod{6}, \\ 3 \pmod{6}, \\ 5 \pmod{6}. \end{cases}$$

Theorems:***Fermat's Little Theorem***

If p is a prime and p does not divide $c \in \mathbb{N}$, then $c^{p-1} \equiv 1 \pmod{p}$.

Chinese Remainder Theorem

If m_1, \dots, m_t are all relatively prime, then for any integers a_1, \dots, a_t the following system of congruences has a unique solution mod $m_1 \dots m_t$:

$$\begin{cases} x \equiv a_1 \pmod{n_1}, \\ x \equiv a_2 \pmod{n_2}, \\ \vdots \\ x \equiv a_t \pmod{n_t}. \end{cases}$$

Cryptography: RSA Secure Encoding-Decoding Algorithm

1. In order to encode a natural number M , choose two different primes p and q , then choose $s \in \mathbb{N}$ as relatively prime to $(p-1)$ and to $(q-1)$. Let $r = pq > M$. Compute the encoding E as $E \equiv M^s \pmod{r}$, $0 \leq E < r$. You only need the values of r and s to encode M .

2. In order to decode M given E , p , q and s , solve the congruences:

$$\begin{cases} M \equiv E^a \pmod{p}, & as + x(p-1) = 1, \\ M \equiv E^b \pmod{q}, & bs + y(p-1) = 1. \end{cases}$$

Proof: using Fermat's Little Theorem and the Chinese Remainder Theorem. Security is guaranteed by the practical difficulty of decomposition of a large integer r into primes p and q .