# Discrete Optimization 2010
# Lecture 6
# Total Unimodularity & Matchings

Marc Uetz
University of Twente

m.uetz@utwente.nl

# Outline

1. Total Unimodularity
2. Matching Problem

# Question

Under what conditions on $A$ and $b$ is it true that all vertices of $P = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$ happen to be integer?

Answer: If $A$ is totally unimodular, and $b$ integer

# (Totally) Unimodular Matrices

## Definition

1. An integer square matrix $B \in \mathbb{Z}^{n \times n}$ is unimodular if $\det(B) = \pm 1$
2. An integer matrix $A \in \mathbb{Z}^{m \times n}$ is totally unimodular (TU) if each square submatrix $B$ of $A$ has $\det(B) \in \{0, \pm 1\}$.

Some easy facts:

- entries of a TU matrix are $\{0, \pm 1\}$ by definition ($1 \times 1$ submatrices)
- if $A$ is TU, adding or deleting a row or column vector $(0, \dots, 1, \dots, 0)$, the result is again TU
- if $A$ is TU, multiplying any row/column by $-1$, the result is again TU

# Motivation

If all vertices of $\{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$ are integer, for any $c$ the linear program

$$\begin{aligned} \max \ & c^t x \\ \text{s.t. } & Ax \leq b, \ x \geq 0 \end{aligned}$$

(if not infeasible or unbounded) has an integer optimal solution as the set of optimal solutions of $P$ is a face of $\{x \mid Ax \leq b, x \geq 0\}$

In that case, we can even solve integer linear programming problems by solving an LP only (say with Simplex), because an optimum solution (if it exists) also occurs at a vertex which happens to be integer.

# Integrality of Linear Equality Systems

**Theorem**

Let $B$ be an integer square matrix that is nonsingular (that is, $\det(B) \neq 0$), consider system $Bx = b$. Then $x$ is integer for any integer right-hand-side $b$ if and only if $B$ is unimodular.

"if" $x = B^{-1}b$, and by Cramer's rule (Linear Algebra):

$$x_j = \frac{\det(B^j)}{\det(B)}$$

with $B^j = B$, but with *jth* column of $B$ replaced by $b$

"only if" $x = B^{-1}b$ integer for all $b$, also for $b^t = (0, \ldots, 1, \ldots, 0)$
such an $x$ exactly equals a column of $B^{-1}$
so all columns of $B^{-1}$ are integer, and so is $\det(B^{-1})$
but $\det(B)\det(B^{-1}) = 1$, both integer, so both are $\pm 1$

# TU Matrices and Integrality

Define for $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

$$P(A) = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$$

**Theorem**

If $A$ is TU and $b$ integer, then $P(A)$ has integer vertices only.

# Proof

$$P(A) = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$$

If $P(A) = \emptyset$ nothing to prove. We write
$P(A) = \{x \mid \begin{bmatrix} A \\ -E \end{bmatrix} x \leq \begin{bmatrix} b \\ 0 \end{bmatrix}\}$. As $A$ is TU, so is $\begin{bmatrix} A \\ -E \end{bmatrix}$.

A vertex $x$ of $P(A)$ exists (see Exercise), and is defined by $n$ linearly independent inequalities of this system, so by $A^o x = b^o$ for non-singular subsystem $A^o, b^o$ of $\begin{bmatrix} A \\ -E \end{bmatrix} x \leq \begin{bmatrix} b \\ 0 \end{bmatrix}$.

As $A^o$ non-singular and TU, $\det(A^o) = \pm 1$, so $x = (A^o)^{-1} b^o$ is integer. □

# The whole story: Hoffman-Kruskal Theorem

$$P(A) = \{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$$

We proved sufficiency:

**Theorem**

If $A$ is TU and $b$ integer, then $P(A)$ has integer vertices only.

It also holds necessity (see Literature, Theorem 6.25):

**Theorem**

If $P(A)$ has integer vertices for all integer $b$, then $A$ is totally unimodular.

# Sufficient Condition Total Unimodularity

**Theorem**

A matrix $A$ is totally unimodular if no more than 2 nonzeros are in each column, and if the rows can be partitioned into two sets $I_1$ and $I_2$ such that

1. If a column has two entries of the same sign, their rows are in different sets of the partition
2. If a column has two entries of different sign, their rows are in the same set of the partition

Proof: By induction on the size of the square submatrices □

# Sufficient Condition Total Unimodularity

$$
\begin{array}{c}
I_1 \\
\\
I_2
\end{array}
\quad
\left(\begin{array}{ccccccc}
 & & & 1 & & & \\
1 & 1 & & & & -1 & \\
 & & & & 1 & 1 & \\
 & & & & -1 & & \\
\hline
1 & & & 1 & & & 1 \\
 & 1 & 1 & & & & -1
\end{array}\right)
$$

# Minimum Cost Flows

**Theorem**

The node-arc incidence matrix of any directed graph $G = (V, E)$ is totally unimodular.

Proof: Rows have one $+1$ and one $-1$, so take $I_1 = V$, $I_2 = \emptyset$ □

**Consequence**

The linear program for Min-Cost Flow always has integer optimal solutions, as long as capacities $u_{ij}$ and balances $b(i)$ are integer.

The dual linear program always has integer optimal solution, as long as the costs $c_{ij}$ are integer.

# Outline

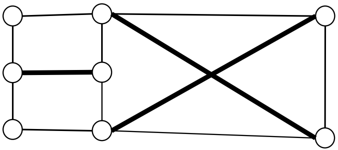
1. Total Unimodularity
2. Matching Problem

# The Matching Problem

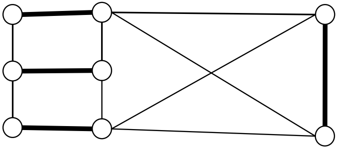
**Definition**

A matching in an undirected graph $G = (V, E)$ is a set $M \subseteq E$ of pairwise non-incident edges.

Given an undirected graph $G = (V, E)$, a maximum matching is one with maximal cardinality. Perfect matching: $|M| = |V|/2$



maximal

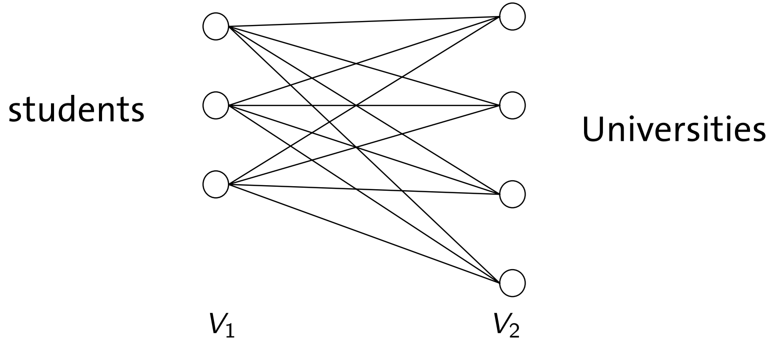


maximal & maximum

# Special Case: Bipartite Matching

## Special Case

Matching in bipartite graphs $G = (V_1, V_2, E)$, where $E \subseteq V_1 \times V_2$.



# Bipartite Graphs

**Theorem**

Graph $G = (V, E)$ is bipartite if and only if $G$ contains no odd cycle.

Proof: (w.l.o.g. assume $G$ connected)

Necessity: Trivially, a bipartite graph has no odd cycle.

Sufficiency: Given $G$ with no odd cycle. Pick $v_o \in V$. Define

- $V_1 = \{v \in V \mid \text{dist}(v_o, v) \text{ even}\}$
- $V_2 = \{v \in V \mid \text{dist}(v_o, v) \text{ odd}\}$

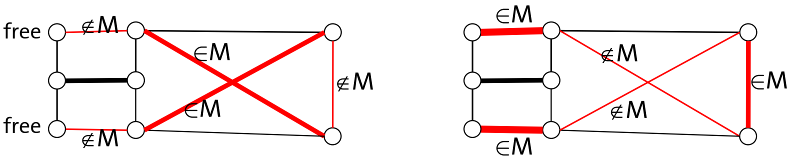
Then $V_1 \cup V_2 = V$ is a bipartition with no edges within $V_1$, $V_2$.

$\square$

# Matching Algorithms: Basic Ideas

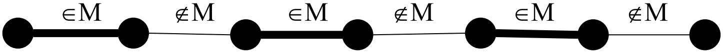
- matchings are an independence system but no matroid
- greedy algorithm yields maximal matching, but needn't be maximum
- for $M$ = maximal matching, and $M^*$ = maximum matching, $|M| \geq \frac{1}{2}|M^*|$ (Exercise)

Idea for matching algorithm: Start with maximal matching, look for augmenting paths

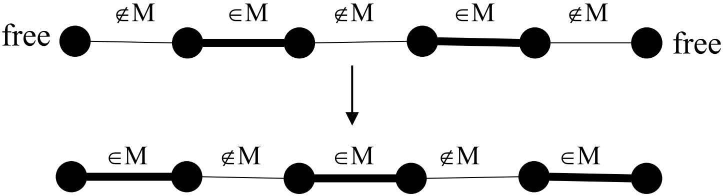


# Alternating & Augmenting Paths

alternating:




augmenting:



# Alternating Path Theorem

Observation: Let $M$ be a matching of $G = (V, E)$, and let $P$ be an augmenting path (for $M$), then $M \oplus P$ is a matching with one edge more.

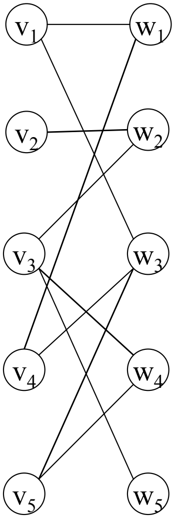
$M \oplus P = M \cup P \setminus M \cap P$

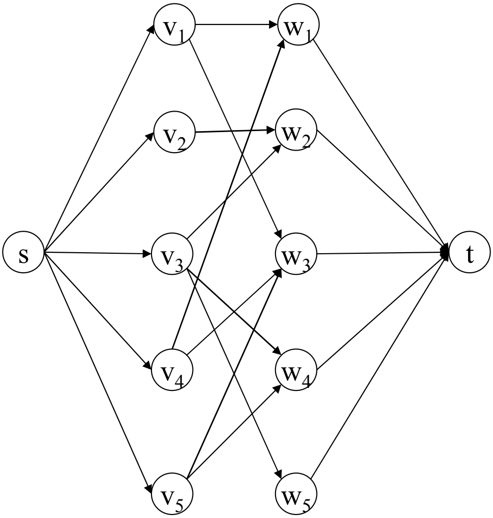
**Theorem (Berge 1957)**

$M$ is a maximum matching if and only if there are no augmenting alternating paths (for $M$).

Proof: Necessity is clear, sufficiency: Assume matching $M'$ larger than $M$ and consider $M \oplus M'$. Show that this contains an augmenting alternating path for $M$. (see page 129, reader) □

# Maximum Bipartite Matching and Maximum Flow



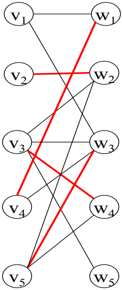




all capacities equal 1

# Maximum Bipartite Matching and Maximum Flow

## Augmenting Path Algorithm for MaxFlow

- Flow values are either 0 or 1
- flow 1 on $(v, w) \Leftrightarrow$ edge $\{v, w\} \in M$
- any feasible flow yields matching (why?)
- flow value $v$ = size of the matching $|M|$
- Computation time $O(\, v(n+m)\,) \in O(\, nm\,)$




## Question

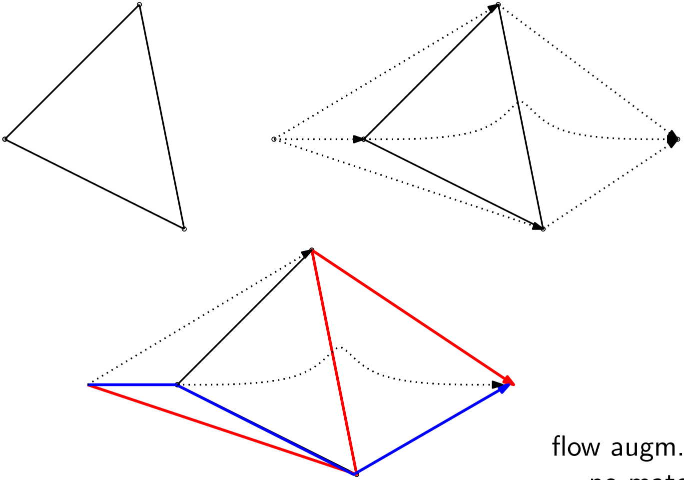
What are the flow augmenting paths?

- red arc: flow 1, backward arc in residual graph
- black arc: flow 0, forward arc in residual graph

So, flow augmentation = augmenting alt. path

# Problem for Non-Bipartite Graphs

Odd cycles:



flow augm. path
no matching!

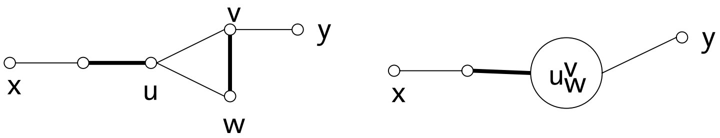
# Edmonds' Breakthrough

**Theorem (Edmonds 1965)**

There exists a polynomial time algorithm for finding a maximum matching in any (also non-bipartite) graph. O( $n^3$ ) time

*Paths, Trees, and Flowers*, Canad. J. Math. 17 (1965)

Main idea: 'Shrink the blossoms (odd cycles)'




Why breakthrough? (first polynomial time algorithm for a problem where the constraint matrix isn't TU)

# Matching: Integer Linear Programming Formulation

Recall $\delta(v)$ = edges incident with $v$

**$x_e = 1$ if edge $e$ is in $M$**

$$\begin{aligned} \max \quad & \sum_{e \in E} x_e & \\ \text{s.t.} \quad & \sum_{e \in \delta(v)} x_e \leq 1 & v \in V \\ & x_e \in \{0,1\} & e \in E \end{aligned}$$

With $A$ = node-edge incidence matrix, this is

$$\begin{aligned} \max \quad & \mathbf{1} \cdot x \\ \text{s.t.} \quad & Ax \leq \mathbf{1} \\ & x \geq 0, \text{integer} \end{aligned}$$

# Matching: Linear Programming Formulation

**Theorem**

Node-edge incidence matrix $A$ of an undirected graph is totally unimodular if (and only if) $G$ is bipartite. (Exercise)

**Consequence?**

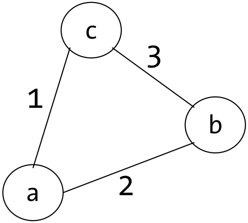
The linear program (LP) for the matching problem

$$\begin{aligned} \max \ & \mathbf{1} \cdot x \\ \text{s.t. } & Ax \leq \mathbf{1} \\ & x \geq 0 \end{aligned}$$

always has an integer optimal solution (a matching), if $G$ bipartite.

And as before, the same holds for the dual LP ... what is the dual?

# Matching & Node Cover




**Matching**: max $\mu = x_1 + x_2 + x_3$

$$x_1 + x_2 \leq 1$$
$$x_2 + x_3 \leq 1$$
$$x_1 + x_3 \leq 1$$
$$x \geq 0, \text{integral}$$

**Node cover**: min $\nu = y_a + y_b + y_c$

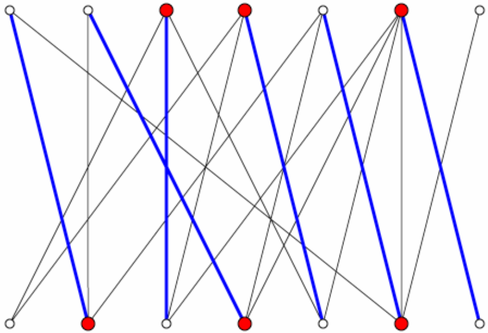
$$y_a + y_c \geq 1$$
$$y_a + y_b \geq 1$$
$$y_b + y_c \geq 1$$
$$y \geq 0, \text{integral}$$

We know $\mu \leq \nu$ (why?)

# Matching & Node Cover in Bipartite Graphs

**Theorem (Kőnig 1931)**

In any bipartite graph $G$, the size of a maximum matching equals the size of a minimum node cover.



# Proof of Kőnig's Theorem

The Linear Programming relaxations of Matching and Node Cover are the following primal dual pair:

**Matching**

$$\begin{aligned} \max \quad & \mathbf{1} \cdot x \\ \text{s.t.} \quad & Ax \leq \mathbf{1} \\ & x \geq 0 \end{aligned}$$

**Node Cover**

$$\begin{aligned} \min \quad & \mathbf{1} \cdot y \\ \text{s.t.} \quad & A^t y \geq \mathbf{1} \\ & y \geq 0 \end{aligned}$$

Bipartite graph $\Rightarrow$ matrix $A$ is TU $\Rightarrow$ $\exists$ optimal primal-dual pair that is integer, and this is the desired matching/node-cover. □

# Square Submatrix

$$\begin{vmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{vmatrix} \curvearrowright \begin{vmatrix} 1 & 1 \\ 0 & 1 \end{vmatrix}$$

▸ Back