Matchings:

\( G = (V, E) \) undirected

A **matching** is a set of **disjoint** edges

An **augmenting path** for a matching

* \( M \) is a path with
  * odd length
  * first, last vert not covered by \( M \)
  * alternate edges in \( M \)

**Examples:**

\[ \begin{array}{c}
\circ & \circ & \circ & \circ & \circ & \circ & \circ \\
\end{array} \]

\[ \begin{array}{c}
\circ & \circ & \circ & \circ & \circ & \circ & \circ \\
\end{array} \]

\[ \begin{array}{c}
\circ & \circ & \circ & \circ & \circ & \circ & \circ \\
\end{array} \]
Define symmetric difference $\Delta$ by

$$A \Delta B = (A - B) \cup (B - A)$$

$= \{ \text{everything in one, but not both} \}$

If $P$ is an $M$-augmenting path,

then $M' = M \Delta E_P$

is a matching with $|M'| = |M| + 1$

Thus: $G = (V,E)$ a graph, $M$ a matching. Then either $M$ is a

matching of max size, or there exists an $M$-augmenting path.
Proof:

If \( M \) is a max-size matching, cannot have \( M \)-augmenting \( P \) because \( M \cup \text{EDP} \) would be bigger than \( M \).

If \( M' \) is a matching, larger than \( M \), consider

\( G' = (V, M' \cup M) \)

This has max degree 2.

So each component is a path (perhaps length 0) or a circuit.

\(|M'| > |M|\) so one component must have more \( M' \) edges than \( M \) edges. This is an \( M \)-augmenting path. \( \square \)
Maximum size bipartite matching

G = (V, E), bipartite

M matching

Input

Output: M', such that |M'| > |M|

Alg:
- G has color classes U, W.
- D_M is directed.
  - vertices are vertices of G
  - edges are edges of G
    e ∈ M, e goes W → U
    e ∉ M, e goes U → W
- U_M is elements in U not covered by M
  \[ W_M \]
- find a directed path from $U_m$ to $W_m$ in $D_m$
- this is $M$ augmenting, so gives a matching larger than $M$.

Weighted bipartite matching.
(Hungarian alg.)
- let each edge has a weight $w(e)$
- we have $G = (V,E)$, color classes $U, W$; $w : E \to \mathbb{Q}$

Method
- start with $M = \emptyset$
- construct $D_m$, directed by
- orienting each edge in $M$ to go $W \to U$, with length $= w(e)$
- all others go $U \to W$, length $= -w(e)$
Write $U_m$ for $u$ verts not covered by $M$

$W_m \leftarrow W$

- find shortest $U_m - W_m$ path (if it exists), say $P$

Form $M \rightarrow M \uplus EP$

and iterate.

Stop when no $P$ can be found.

Thus: call a matching extreme if it has max weight among all size $|M|$ matchings. Each $M$ found by this method is extreme.
Proof: (Induction)

1. True for $M = \emptyset$

2. Suppose $M$ is extreme, $P$ and $M'$ from matching alg. show $M'$ is extreme.


4. $|N| > |M|$, so $M \cup N$ has a component $Q$ that is $M$-augmenting.

5. $P$ is shortest such, so $\ell(P) \leq \ell(Q)$

6. $N \triangle Q$ is a matching, $|N \triangle Q| = |M| 

7. $\Rightarrow$ Why

8. but $M$ is extreme, so $w(M) > w(N \triangle Q)$

9. $\Rightarrow$ Why

10. $w(N) = w(N \triangle Q) - \ell(Q) \leq w(M) - \ell(P) = w(M')$

11. So $M'$ is extreme

$\square$
linearly independent rows of $A$ and $W'$ gives a maximum-size set of linearly independent columns of $A$. Then the $U' \times W'$ submatrix of $A$ is nonsingular, hence of nonzero determinant. It implies (by the definition of determinant) that $G$ has a matching covering $U' \cup W'$.

(Related work includes Perfect and Pym [1966], Pym [1967], Brualdi [1965a,1971b], and Mirsky [1969].)

16.7. Further results and notes

16.7a. Complexity survey for cardinality bipartite matching

Complexity survey for cardinality bipartite matching ($*$ indicates an asymptotically best bound in the table):

\[ O(nm) \quad \text{König [1931], Kuhn [1955b]} \]

\[ O(\sqrt{n}m) \quad \text{Hopcroft and Karp [1971,1973], Karzanov [1973a]} \]

\[ * O(n^{1.5}) \quad \text{Ibarra and Moran [1981]} \]

\[ O(n^{1.5} \frac{m}{\log n}) \quad \text{Alt, Blum, Mehlhorn, and Paul [1991]} \]

\[ * O(\sqrt{n}m \log_2 (n^{2}/m)) \quad \text{Feder and Motwani [1991,1995]} \]

Here \( \omega \) is any real such that any two \( n \times n \) matrices can be multiplied by \( O(n^\omega) \) arithmetic operations (e.g. \( \omega = 2.376 \)).

Goldberg and Kennedy [1997] described a bipartite matching algorithm based on the push-relabel method, of complexity \( O(\sqrt{n}m \log_2 (n^{2}/m)) \). Balinski and Gonsalez [1991] gave an alternative \( O(nm) \) bipartite matching algorithm (not using augmenting paths).

16.7b. Finding perfect matchings in regular bipartite graphs

By König's matching theorem, each \( k \)-regular bipartite graph has a perfect matching (if \( k \geq 1 \)). One can use the regularity also to find quickly a perfect matching. This will be used in Chapter 20 on bipartite edge-colouring.

First we show the following result of Cole and Hopcroft [1982] (which will not be used any further in this book):

**Theorem 16.9.** A perfect matching in a regular bipartite graph can be found in \( O(m \log n) \) time.

**Proof.** We first describe an \( O(m \log n) \)-time algorithm for the following problem:

\begin{equation}
(16.9) \quad \text{given: a \( k \)-regular bipartite graph } G = (V,E) \text{ with } k \geq 2, \text{ find: a nonempty proper subset } F \text{ of } E \text{ with } (V,F) \text{ regular.}
\end{equation}
Further notes

17.5. Further results and notes

Complexity survey for maximum-weight bipartite matching (as indicated in the table).