Minimum cost flows:

\[ D = (V, A) \] directed graph
\[ k: A \rightarrow \mathbb{R} \] cost function

for any \( f \),

\[ \text{cost}(f) = \sum_{a \in A} k(a) f(a) \]

\( s, t \in V \)

Minimum cost flow

given these, a capacity \( c \), demand values \( \Phi \), find:

\( d \leq \text{s-t flow} \leq c \), value = \( \Phi \),

with minimum cost

includes:

Find max-value flow \( \leq c \) that has min cost.
We will work with circulations:

**Circulation:**

directed function $f: \text{edges} \to \mathbb{R}$ such that

$$\text{excess}(v) = \sum_{e \in \text{incomimg at } v} f(e) + \sum_{e \in \text{outgoing at } v} f(e) = 0$$

for all verts $v$

**Useful fact:**

**Theorem:** $D = (V, A)$ a directed graph, $f: A \to \mathbb{R}$, then $f$ is a nonnegative linear comb. of at most $|A|$ vectors $X^p$, where $P$ is a directed path or circuit. If $P$ is a path, it starts at a vertex $v$ with $\text{excess}_f(v) < 0$, ends at vertex with $\text{excess}_f > 0$. 
Proof:

1) we add a new edge for every vertex, $u$, to make $\text{excess}_f(u) = 0$ everywhere.

a) add $u$

b) for each $v$ such that $\text{excess}_f(v) > 0$, add $v, u$ with $\text{flow} = \text{excess}_f(v)$.

c) for each $v$ s.t. $\text{excess}_f(v) < 0$, add $u \rightarrow v$ with $\text{flow} = -\text{excess}_f(v)$

d) we now have $\text{excess}_f(w) = 0 \forall w \rightarrow [\text{Why?}]$

2) Now prove for $\text{excess}_f(w) = 0$.

$A' = \{a \mid f(a) > 0\}$ injection

$A' \neq \emptyset$ (otherwise easy)
so $A'$ contains a directed circuit, $c$.

Let $I$ be min $\{f(a) \mid a \in C \}$.

- we have one $x^c$, which is $x^c$
- set $f' = f - e$ and go again

Notice: we have removed at least one edge, so there can be no more than $|A| - 1$ $x^c$'s

- this $x^c$ appears with non-negative coefficients
- the construction yields the constraints on paths

This theorem reads, for circulations:

Each non-negative circulation is a non-negative linear combination of incidence vectors of directed circuits.
back to minimum cost. We work with circulations

Formalize our approach to flow augmenting paths

for a digraph, $D = (V, A)$ with $f : A \rightarrow \mathbb{R}$, capacity function $c$, demand function $d$, define a residual graph:

$D_f = (V, A_f)$

$A_f = \{ a \mid a \in A, f(a) < c(a) \}$

$\cup$

$\{ a^{-1} \mid a \in A, f(a) > d(a) \}$

reverse the direction of $a$. 

edges which can take more toward flows

edges where we can reduce flow i.e. edge can take more backward flow.
Notation:

- extend cost $k(a)$ to $A^-$ by
  $$k(a^-) = -k(a).$$

- directed circuit $C$ in $D$
gives an undirected circuit $c$ in $D$

- $X(a) = \begin{cases} 1 & \text{if } C \text{ traverses } a^- \\ -1 & \text{if } C \text{ traverses } a \\ 0 & \text{otherwise} \end{cases}$

- $f$ is feasible if $d \leq f \leq c$.

Theorem: $D = (V,A)$ digraph; $d, c, k : A \rightarrow K$

$f : A \rightarrow K$ is a feasible circulation.

$f$ has min cost among all feasible circs $\iff$ circuit of $D^+$ has non-neg cost.
Proof:

\[\implies \text{let } C \text{ be a directed circuit of negative cost. Then there is some } x \text{ such that } f + \varepsilon x \text{ is feasible. But } f + \varepsilon x \text{ has lower cost than } f\]

\[\iff \text{Suppose each directed circuit in } D_f \text{ has non-negative cost. Let } f' \text{ be any feasible circ. Then } f' - f \text{ is a circulation, so}
\]

\[f' - f = \sum_{j=1}^{m} \lambda_j x_j^c \]

\[\lambda_j > 0 \]

\[\text{for some directed circuits } c_j \text{ in } D_f \text{ hence}
\]

\[\text{cost}(f') - \text{cost}(f) = \text{cost}(f' - f)
\]

\[= \sum_{j} \lambda_j k(C_j) > 0\]
Algorithm to improve circulation

- Choose a negative cost directed circuit $C \in D$ and reset

  $f \leftarrow f + \alpha \cdot x^c$

  where $\alpha$ is max subject to $k \cdot f \leq c$.

- If there is no such circuit, we are finished.

Notes:

- arbitrary choice of circuits gives exponential alg.

- choose $C$ of minimum mean cost

  where mean cost is

  \[
  \frac{k(c)}{|C|}
  \]

  gives strongly polynomial time alg.
### 12.5. Min-max relations for minimum-cost flows and circulations

Here $B = \mathbb{R}$ for integer $q$.

<table>
<thead>
<tr>
<th>$c_{m_{j}}$</th>
<th>$g(x)$</th>
<th>$f(x)$</th>
<th>$h(x)$</th>
<th>$i(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[986, 1989]$</td>
<td>$((\mu', v', u, w') + c_{m_{j}} \cdot SP \cdot b_{m_{j}})$</td>
<td>$O$</td>
<td>$*$</td>
<td>$*$</td>
</tr>
<tr>
<td>$[986, 1989]$</td>
<td>$((\mu', v', u, w') + c_{m_{j}} \cdot SP \cdot b_{m_{j}})$</td>
<td>$O$</td>
<td>$*$</td>
<td>$*$</td>
</tr>
<tr>
<td>$[986, 1989]$</td>
<td>$((\mu', v', u, w') + c_{m_{j}} \cdot SP \cdot b_{m_{j}})$</td>
<td>$O$</td>
<td>$*$</td>
<td>$*$</td>
</tr>
<tr>
<td>$[986, 1989]$</td>
<td>$((\mu', v', u, w') + c_{m_{j}} \cdot SP \cdot b_{m_{j}})$</td>
<td>$O$</td>
<td>$*$</td>
<td>$*$</td>
</tr>
</tbody>
</table>

### 12.6. Complementary survey for minimum-cost circulation

<table>
<thead>
<tr>
<th>$c_{m_{j}}$</th>
<th>$g(x)$</th>
<th>$f(x)$</th>
<th>$h(x)$</th>
<th>$i(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[986, 1989]$</td>
<td>$((\mu', v', u, w') + c_{m_{j}} \cdot SP \cdot b_{m_{j}})$</td>
<td>$O$</td>
<td>$*$</td>
<td>$*$</td>
</tr>
<tr>
<td>$[986, 1989]$</td>
<td>$((\mu', v', u, w') + c_{m_{j}} \cdot SP \cdot b_{m_{j}})$</td>
<td>$O$</td>
<td>$*$</td>
<td>$*$</td>
</tr>
<tr>
<td>$[986, 1989]$</td>
<td>$((\mu', v', u, w') + c_{m_{j}} \cdot SP \cdot b_{m_{j}})$</td>
<td>$O$</td>
<td>$*$</td>
<td>$*$</td>
</tr>
<tr>
<td>$[986, 1989]$</td>
<td>$((\mu', v', u, w') + c_{m_{j}} \cdot SP \cdot b_{m_{j}})$</td>
<td>$O$</td>
<td>$*$</td>
<td>$*$</td>
</tr>
</tbody>
</table>

### 12.7. Further results and notes

#### Theorem 12.7

For the minimum-cost circulation problem, the following properties hold:

1. If $x$ is a feasible solution, then $x$ is optimal if and only if $x$ is a primal-dual solution of the minimum-cost circulation problem.
2. If $x$ is a feasible solution, then $x$ is optimal if and only if $x$ is a primal-dual solution of the minimum-cost circulation problem.

#### Corollary 12.7

If $x$ is a feasible solution, then $x$ is optimal if and only if $x$ is a primal-dual solution of the minimum-cost circulation problem.
Matchings:

\[ G = (V, E) \text{ undirected} \]

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:

\[ \text{not covered by } M \]
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 $\text{MDEP}$ 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. $\Box$
Maximum size bipartite matching

(we did this as flow already)

\[ G = (V, E) \text{, bipartite} \]

Input \( M \) matching

Output: \( M' \), such that \( |M'| > |M| \)

Alg:
- \( G \) has color classes \( U, W \).
- \( D_M \) is directed.
  - verts are verts of \( G \)
  - edges are edges of \( G \)
    - \( e \in M \), \( e \) goes \( W \rightarrow U \)
    - \( e \notin M \), \( e \) goes \( U \rightarrow W \)
- \( U \cap M \) is elements in \( U \) not covered by \( M \)

\[ W_M \]
find a directed path from $V_m$ to $W_m$ in $D_m$

- this is $M$ augmenting, so gives a matching larger than $M$.

**Weighted bipartite matching.**  
(Hungarian alg.)

- set each edge has a weight $w(e)$
- we have $G = (V, E)$, color classes $U$, $W$, $w : E \to \mathbb{R}$

**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 \( M \)

\[ W_m \quad "W" \]

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

form \( M \rightarrow M + EP \)
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

3. Consider any \( N \), a matching, extreme, \( |N| = |M| + 1 \).

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

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

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

\[ \Rightarrow |\text{Why}| \leq \]

7. But \( M \) is extreme, so \( w(M) > w(N \triangle Q) \)

\[ \Rightarrow |\text{Why}| \leq \]

8. \( w(N) = w(N \triangle Q) - l(Q) \leq w(M) - l(P) = w(M') \)

9. So \( M' \) is extreme

\[ \square \]
It suffices to show that each vertex cover \( C \) has \( |C| \geq |U| \). This indeed is the case, since \( N(U \setminus C) \subseteq C \setminus W \), and hence
\[
|C| = |C \cap U| + |C \cap W| \geq |C \cap U| + |N(U \setminus C)| \geq |C \cap U| + |U \setminus C| = |U|.
\]

This can be extended to general subsets of \( V \). First, Hoffman and Kuhn [1956b] and Mendelsohn and Dulmage [1958a] showed:

**Theorem 16.8.** Let \( G = (V, E) \) be a bipartite graph with colour classes \( U \) and \( W \) and let \( R \subseteq V \). Then there exists a matching covering \( R \) if and only if there exist a matching \( M \) covering \( R \cap U \) and a matching \( N \) covering \( R \cap W \).

**Proof.** Necessity being trivial, we show sufficiency. We may assume that \( G \) is connected, that \( E = M \cup N \), and that neither \( M \) nor \( N \) covers \( R \). This implies that there is a \( u \in R \cap U \) missed by \( N \) and a \( w \in R \cap W \) missed by \( M \). So \( G \) is an even-length \( u - w \) path, a contradiction, since \( u \in U \) and \( w \in W \).

(This theorem goes back to theorems of F. Bernstein [1898] p. 103), Banach [1924], and Knaster [1927] on injective mappings between two sets.)

Theorem 16.8 implies a characterization of sets that are covered by some matching:

**Corollary 16.8a.** Let \( G = (V, E) \) be a bipartite graph with colour classes \( U \) and \( W \) and let \( R \subseteq V \). Then there is a matching covering \( R \) if and only if \( |N(S)| \geq |S| \) for each \( S \subseteq R \cap U \) and for each \( S \subseteq R \cap W \).

**Proof.** Directly from Theorems 16.7 and 16.8.

It also gives the following exchange property:

**Corollary 16.8b.** Let \( G = (V, E) \) be a bipartite graph, with colour classes \( U \) and \( W \), let \( M \) and \( N \) be maximum-size matchings, let \( U' \) be the set of vertices in \( U \) covered by \( M \), and let \( W' \) be the set of vertices in \( W \) covered by \( N \). Then there exists a maximum-size matching covering \( U' \cup W' \).

**Proof.** Directly from Theorem 16.8: the matching found is maximum-size since \( |U'| = |W'| = \nu(G) \).

**Notes.** These results also are special cases of the exchange results on paths discussed in Section 9.6c. Perfect [1966] gave the following linear-algebraic argument for Corollary 16.8b. Make a \( U \times W \) matrix \( A \) with \( a_{u,w} = x_{u,w} \) if \( uw \in E \) and \( a_{u,w} := 0 \) otherwise, where the \( x_{u,w} \) are independent variables. Let \( U' \) be any maximum-size subset of \( U \) covered by some matching and let \( W' \) be any maximum-size subset of \( W \) covered by some matching. Then \( U' \) gives a maximum-size set of 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 [1969b,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):

|\( O(nm) \) (König [1931], Kuhn [1955b]) | \( O(\sqrt{n}m) \) (Hopcroft and Karp [1971,1973], Karzanov [1973a]) |

* \( O(n^{\omega-\epsilon}) \) (Ibarra and Moran [1981]) |

* \( O(n^{1/3} \sqrt{\frac{m}{\log n}}) \) (Alt, Blum, Mehlhorn, and Paul [1991]) |

* \( O(\sqrt{n}m \log n(n^2/m)) \) (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 Gonzales [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:

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

17.9. Complexity survey for maximin-weight bipartite matching

17.9.1. Complexity survey

The complexity of the maximin-weight bipartite matching problem is a topic of ongoing research. Various algorithms have been developed to solve this problem, each with its own advantages and disadvantages. The most widely used algorithms include the Hungarian method and the Kuhn-Munkres algorithm. Both of these methods are based on the concept of finding a perfect matching in a bipartite graph.

The Hungarian method, also known as the Kuhn-Munkres algorithm, is a popular technique for solving the problem. It works by iteratively improving the solution until an optimal matching is found. The algorithm is efficient, with a time complexity of O(n^3), where n is the size of the graph.

The Kuhn-Munkres algorithm, on the other hand, is a greedy approach that can be used to solve the problem. It works by selecting the maximum weight edge that does not create a cycle, and then removing it from the graph. This process is repeated until all the edges have been used.

In addition to these algorithms, there are also heuristic methods that can be used to approximate the solution. These methods are often used when the exact solution is not required, or when the problem is too large to be solved exactly.

Overall, the complexity of the maximin-weight bipartite matching problem is an active area of research, with new algorithms and methods being developed constantly.